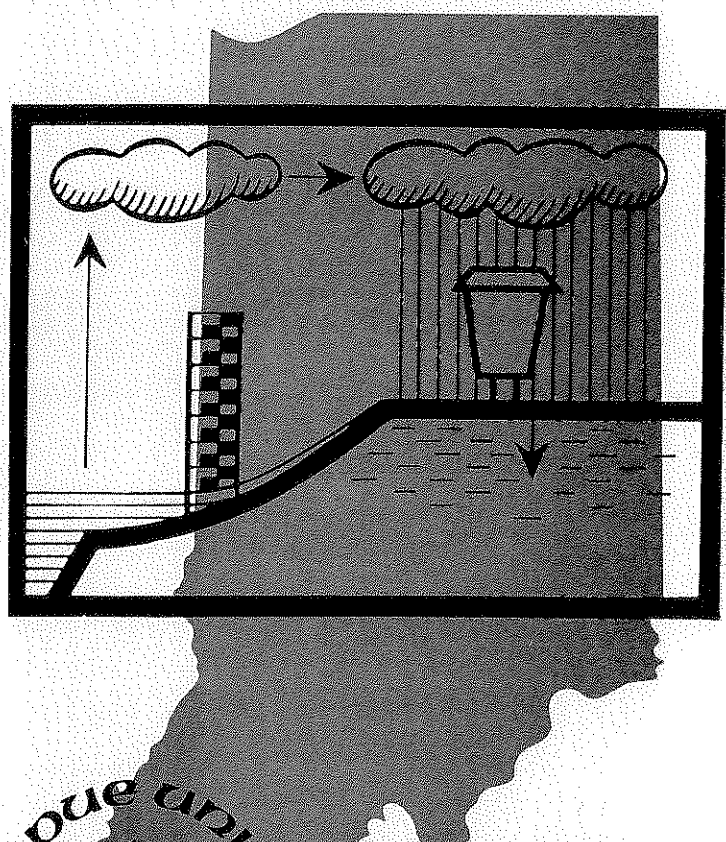
TECHNICAL REPORT NO. 76

WABASH RIVER SYSTEMS MODEL - TERMINAL REPORT PHASE I



by

G. H. Toebes

C. Rukvichai

T. A. Sullivan

July 1976

PURDUE UNIVERSITY
WATER RESOURCES RESEARCH CENTER
WEST LAFAYETTE, INDIANA

Water Resources Research Center
Purdue University
West Lafayette, Indiana

WABASH RIVER SYSTEMS MODEL - - TERMINAL REPORT PHASE I

by

G. H. Toebes
C. Rukvichai
T. A. Sullivan

Period of Investigation: January 1976-June 1976
Aggreement No. 14-34-6015
OWRT-A-026-IND

Purdue University Water Resources Research Center
Technical Report 76
July 1976

PWRRC REPORT 76

WABASH RIVER SYSTEMS MODEL - TERMINAL REPORT PHASE I

TABLE OF CONTENTS

WABASH RIVER SYSTEMS MODEL - TERMINAL REPORT PHASE I

(OWRR-A-013-IND, OWRR-A-016-IND, OWRR-A-026-IND)

PREFACE AND ACKNOWLEDGEMENT

During the 1969-1974 period a series of projects sponsored by the Purdue Water Resources Research Center (PWRRC), led to building a computer simulation model for the Upper Wabash River and Reservoir System. This work, done in the School of Civil Engineering, Purdue University, was interrupted in 1974 because of unavailability of personnel.

That personnel situation persists to date (January 1976). It is advisable, therefore, to formally deactivate the project while insuring that the investment that has been made remains available. This investment has the form of a software system consisting of: (a) a data bank and its access routines; (b) a program library representing the simulation model components; (c) supporting information.

Maintaining the availability of the software system is of value only if further use can be identified and if the expected benefits thereof exceed the expected costs. Considering that equation is one of the purposes of this terminal report. Doing so requires taking stock of what was accomplished and what remains undone. This then is another aim of the present report.

The work accomplished resulted in a software system that permits one to generate detailed information for surface water project planning and management questions for the 8000 mi^2 Upper Wabash basin in Indiana. The model yields daily reservoir surface elevations, volumes, river flows, and river stages in response to hydrologic occurrences and to the manner in which storage reservoirs are operated. It permits one to make a variety of statistical evaluation studies using simulation results. In addition the question of project benefits was considered, albeit quite partially.

The creation of software for daily simulation of an 8000 mi^2 basin containing up to five storage reservoirs that would be suitable for actual agency use requires resources that considerably exceed those available to the above referenced projects. Aside from this, a field-operational model cannot and should not be built in a university setting. While, as a consequence, many parts of the work remain unfinished, considerable progress

was made. This could not have been the case were it not for the very willing cooperation of many staff members in state and federal agencies. Particular acknowledgements are due to Dr. T. P. Chang of the Indiana Stream Pollution Control Board; Bill Andrews, Bob Jackson, Dr. Prasad, Jim Russel, John Simpson, and Victor Wenning all of the Indiana Department of Natural Resources; Colonel Fiala, Noah Whittle, Bill Leagan and Carl Fleener and others of the Louisville District Office, Corps of Engineers, Kentucky; Malcolm Hale and Dick Hogart of the USGS District Office, Indianapolis, Indiana; and Karl Relya, National Weather Service, River Forecast Center, Cincinnati, Ohio. In regard to help for work done on the evaluation of reservoir recreation benefits we like to thank John Costello, Thomas Huck, Bill Walters, Clay McDermiott, Tom Huck, and Dave Griffith all of the Indiana Department of Natural Resources; and John Gleidt and Glenn Bayes of the Louisville District Office, Corps of Engineers.

The project owes much to post-doctoral fellows, Drs. T. P. Chang and S. Tuffuor; and to David Lyman and Tom Sullivan, software managers. It was sponsored and administered by the Purdue Water Resources Research Center, Dr. Dan Wiersma, Director; G. H. Toebes was principal investigator.

The current and final phase of the work reported herein received financial sponsorship from OWRT (agreement no. 14346015) via the Purdue Water Resources Research Center, Dr. D. Wiersma, Director, and from the School of Civil Engineering, Purdue University, Dr. J. F. McLaughlin, Head, and from the Corps of Engineers via the Louisville District Office, and supervised by the Cincinnati's Office Hydrology Branch; G. T. Mitchell, Supervisor.

WABASH RIVER SYSTEMS MODEL - TERMINAL REPORT PHASE I

ABSTRACT

A daily simulation model was developed for the Upper Wabash River and Reservoir System in Indiana. The motivation for this work was to apply then (1969) yet novel systems approaches to surface water planning and management facets for a major river in the state of Indiana.

The core of the project was the building of river-reservoir simulation software including reservoir operating policy rules and to do so for a 24-hour time step. The construction of the simulation model, the different operating policies used, a discussion of benefit estimation, and the software documentation have been detailed in five previous reports (see footnote).

While the present report contains a brief summary of the material found in the earlier publications (see Chapter II and III), its major purpose is first of all to record some overall conclusions (see Chapter I). Secondly, it aims to state what related work is advisable and what corresponding benefits may be expected (see Chapters III and IV).

Additional work on programs, particularly on data base organization, access, and editing and documentation was performed. The record is found in Chapters V, VII and IX.

G. H. Toebes and T. P. Chang: "Simulation Model for the Upper Wabash Surface Water System", PUWRRC Techn. Report #27, 100 pg. (July 1972)

T. P. Chang and G. H. Toebes: "Operating Policies for the Upper Wabash Surface Water System", PUWRRC Techn. Report #31, 73 pb. (Dec. 1972)

T. P. Chang and G. H. Toebes: "Initial Results from the Upper Wabash Simulation Model", PUWRRC Techn. Report #33, 89 pg. (March 1973)

K. K. Wolka and G. H. Toebes: "Estimating Reservoir Recreational Visits in Indiana", PUWRRC Techn. Report #48, 99 pg. (June 1974)

S. Tuffuor, D. Lyman, G. H. Toebes: "Upper Wabash Simulation Model: Program Documentation and Extension", PUWRRC Techn. Report #49a, 107 pg. (June 1974)

I. INTRODUCTION

1. GENERAL AND SPECIFIC PROJECT OBJECTIVES

This report concludes the first and principal phase of a research initiated some 6 years ago. The original objectives for the undertaking were:

(a) to gain experience and to advance the *use* of some of the then fairly new systems methodologies;

(b) to promote such use for a regional water resource situation in *Indiana*;

(c) to identify for that region what *additional data* were needed for effective surface water planning and *management*;

(d) to find how effective a systems model can be in providing information relevant to the then pending controversy and public debate about an authorized reservoir;

(e) to provide training and education in water resource systems analysis.

Those stated goals were obviously rather general ones. Their translation into specific objectives during early project phases was as follows:

(1) build a *simulation model* comprising the Upper Wabash River down to Williamsport and incorporating up to five reservoirs which had been authorized;

(2) study and if possible make a geographic extension of the model down to the Ohio River;

(3) *Survey the literature* on river-reservoir simulation modeling and develop any additional concepts needed for the Upper Wabash study;

(4) devise effective ways to *present the information* to potential users;

(5) consider quantitatively the question of *flood control benefits*;

(6) consider quantitatively the question of water based recreation benefits;

(7) provide training and education in water data management and simulation modeling.

(8) consider quantitatively the question of water quality control benefits.

2. IMPLEMENTATION OF OBJECTIVES

It is of interest to ask which of these objectives were fulfilled, which ones were not implemented, what new objectives did in effect evolve, and finally for each of these to state the reasons why. The Table I-1 is an attempt to answer these questions. The Table, furthermore, serves as a compact project overview. It also is a framework for some general conclusions.

ORIGINAL SPECIFIC OBJECTIVES	PERCENT IMPLEMENTED	COMMENTS
1. Build Simulation Model Upper Wabash Basin	75%	75% rather than 100% because available rainfall data were not used in developing systems parameters.
2a. Study Geographic Model Extension	75%	Missing stream flow data, data management, and the need for some type of hierarchic model decomposition were identified as obstacles to geographic model extension.
2b. Implement Geographic Model Extension	5%	Comments under 2a state reasons for not progressing beyond 5%.
3. Provide Training and Education	100%	Two post-doctoral training efforts; one PhD and one MSc. Theses were completed, and as many as seven other students worked on project data acquisition, data management, programming, and model analysis tasks.
4a. Literature Survey (1970)	80%	A good literature survey was accomplished in 1969, 1970. This set the project strategy.
4b. Literature Survey (1970-75)	20%	The literature after 1970 contains new methodology relative to the Upper Wabash Model. These could not be pursued given the need of project termination due to funding constraints.
4c. Publish Project Results	90%	Results published in five Technical Reports and several papers.
5a. Presentation to Agency Staff and Professionals	70%	Technical presentations to staff Ind. Dept. of Natural Resources, Corps of Engineers, and at various local, national and international meetings.
5b. Use of Results by Agency Staff	40%	Work has led to Corps of Engineers funded work for the Green River Basin, Kentucky. Part of work method is used in the design of storage reservoir in Holland.
5c. Presentation to local public	10%	Presentations to individuals party to the Wildcat Creek Controversy. One radio interview. Effect on public opinion considered to be minimal.

6.	*Flood Control Benefits*	*10%*	*A quantitative, independent evaluation of flood control benefit is a project larger than the current effort. A study of Corps of Engineer's evaluation for the Terre Haute reach led to this conclusion.*
7.	*Water Based Recreation Benefits*	*50%*	*A MSc thesis containing estimation of annual reservoir visitation for the Upper Wabash Basin was completed. Completion requires further input data from a project currently commissioned at Indiana University.*

From the Table 1 one concludes that:

(a) tasks in which Universities do specialize, namely education, research, and publication (tasks #1, #3, #4) were implemented satisfactorily;

(b) tasks that involve contact with potential research results users and an effecting of the use of research results by water agencies were more or less successful (tasks #5a, #5b). This is considerably better than the zero percent implementation not infrequently associated with academic projects.

(c) tasks involving dissemination of results to the general public or even to opinion makers were not apparently successful (task #4c). Improvement on this score would require special non-research type efforts in order to compete with refined, well-financed media efforts that largely control the flow of information. For all their good intentions, most individual faculty members can not be expected to work effectively in that direction for lack of training and appropriate channels.

(d) tasks involving very large amounts of field data, are difficult to complete (tasks #2b, #6, #7). Obvious reasons include budget, time, and "it has to be non-routine" constraints. Regular investment and some centralization of diverse efforts are needed to at least somewhat budge these constraints.

II. TECHNICAL SUMMARY OF WORK ACCOMPLISHED

The work accomplished has been detailed in five previous reports. They are referenced as part of the "Abstract" at the beginning of this report. It appears most effective to first briefly state the nature of the total project and then amplify this by giving a summary of what may be found in each of the five publications referred to earlier.

From a computer programming point of view the core of the work accomplished is a software system that may be divided into the following three components:

(a) input programs, incl. data base access and editing programs;

(b) simulation program, i.e. the actual simulation program which, conceptually, may be divided into physical component (rivers, reservoirs, outlet works, etc.) routines and the reservoir operating policy routine;

(c) output evaluation, or also generated data analysis programs; these include the special graphical plotting routines that were developed and a number of statistical analysis routines.

1. *PROJECT AND PROJECT RESULTS - A TECHNICAL STATEMENT*

Digital computer simulation software was constructed to simulate relevant state variables for a surface water system. The selected system comprised an 8,000 sq. mi. river basin in Northern Indiana (Upper Wabash Basin down to Covington) containing up to five reservoirs (some were built, others were authorized). The selected time step was 24 hours. The state variables included the stages and discharges at USGS river gaging stations and the reservoir elevations and volumes for up to five reservoirs. Sketches of the system are found in the Figures I-1 and I-2.

The software system was used to study the stated (Corps of Engineers) reservoir operating policies and to compare them with systems operating policies, i.e. policies which regard the individual reservoirs as systems components useful in the pursuit of quantitatively defined objectives. The regional objectives in use for the real life system are: (a) flood control, (b) recreation, (c) water quality control, and (d) water supply. The corresponding measures of effectiveness are (a) the aggregated amounts of exceedance of flood stages at key locations along the Wabash River as function of return interval, (b) reservoir related visitor days; (c) effect of low flow constraint level on implementing other objectives; (d) not investigated.

Use of the software system in relating various alternative reservoir operating policies to monetary benefit evaluations requires engineering-economic evaluation models. It was found that building models for the evaluation of flood control benefits as function of stage, and hence of management policy, required data which were impracticable to obtain within the framework of the study. Given flood damage - stage information, however, the simulation software can provide the then needed "stages as function of operating policy" information.

The hydrologic systems inputs were based on earlier hydrologic research for the Upper Wabash region. Proportionally sized simultaneous input hydrographs having the form of gamma functions were used as simulation model inputs. This deterministic input permits isolating the quantitative effects of variations in reservoir operating policy.

2. *SUMMARY OF PUWRRC REPORT #27*

A. CONTENTS

The report #27 contains:

(a) an exposition of simulation modeling concepts as a mode of systems analysis;

(b) a motivation for selecting the Upper Wabash Basin down to Covington as the regional extent of the system;

(c) a description of the hydrology, the systems components, and the water management objectives for that region;

(d) a study of river reach routing models as these represent key elements of the total simulation project;

(e) an exposition how the routing coefficients for the Upper Wabash River were computed;

(f) the manner in which the reservoirs were translated into mathematical model components;

(h) the selection of simple deterministic input hydrographs for the total model;

(i) samples of systems simulation results;

(j) a general description of the developed computer programs;

(k) the results of some sensitivity to routing coefficient studies;

(l) the advantages of using automatic plotting routines for the display of model results.

B. RESULTS

Using preliminary results the report #27:

(a) clearly demonstrated the potential of simulation models (when coupled with automatic plotting routines) to yield quantitative, interrelated, and easily portrayed data that reflect the effects of adding or subtracting reservoirs in the Upper Wabash River basin.

(b) similarly it generated data suggesting that changes in man-made controls, i.e. in the reservoir operating rules (tentatively formulated by the Corps of Engineers) could improve the flood control effectiveness of the reservoirs.

C. LIMITATIONS AND CRITIQUE

Also identified in report #27 were a number of limitations inherent to the approach that was adopted. The principal ones are:

(a) in constructing routing models we made the assumption that ungaged side inflows into river reaches were proportional to the areas of the contributing watershed. Furthermore, that their hydrographs were proportional to the river flow hydrographs at the beginning of the river reach in question. These assumptions resulted from *not using precipitation data* in constructing the river reach models. The model errors found their way in the *routing models.* The river reach representations are thus a weak part of the simulation system.

(b) in the longer run a more serious deficiency than poor routing models is that effective use of the built simulation model in a predictive mode is out. This also results from not using precipitation data. Thus weather forecasts cannot be used as input. This confines the simulation essentially to sensitivity analyses. This sensitivity analysis pertains primarily to the effect of adding or subtracting reservoirs and to changes in their operating rules. To a far lesser extent can sensitivity to realistic hydrologic input variations be studied because (in the absence of precipitation studies) it is not know what constitutes realistic hydrologic input.

3. *SUMMARY OF PWRRC REPORT #31*

A. CONTENTS

The report #31 contains:

(a) a discussion of the role of reservoir operating policies in pursuing surface water management objectives;

(b) a discussion and a translation into algorithm form of the Corps of Engineers reservoir operating rules for the Upper Wabash Basin reservoir; these rules do tie together the reservoir operations only to a limited extent;

(c) a discussion on how to formulate a formal systems operating policy in which up to five reservoirs are operated as a unit for the purpose of providing flood control and recreation benefits;

(d) a discussion on how to adapt "reservoir balancing" policies, to minimize a distributed and spatially related set of flood damages. While "reservoir balancing" has been applied in the operation of reservoirs for hydropower, its application to a flood control and recreation reservoir system causes special difficulties.

(e) an explanation of how the total system was decomposed into subsystems consisting of one reservoir or one river reach each and of the formulation of the operating policy for each such sub-system (Figure 10);

(f) the formulation of a sub-system's short-run operating policy. It contains 48 different sets of constraint and decision equations;

(g) a discussion of how the above sub-system policies were combined into a total systems policy for the Upper Wabash. Two balancing approaches were used, namely the DAR (drainage area ratio) and the SVR (storage volume ratio) policies. These systems operating policies are non-elementary, itterative algorithms;

(h) a discussion of "priority selections" that may be introduced into the systems policies to obtain the sensitivity to weighting of non-commensurate project purposes.

B. RESULTS

(a) A method was developed to formulate systems operating policies for reservoirs having flood control and recreation as primary objectives.

(b) An itterative operations algorithm was built; it is applicable to reservoirs both in series and in parallel. The algorithm was needed because any of the systems policies requires solving sequentially a large set of simultaneous constraint equations. This difficulty was resolved by building an "itterative simulation algorithm".

C. LIMITATIONS AND CRITIQUE

The work reported in Report #31 is subject to a number of limitations. These include:

(a) a simplification of the long-range policy or "guide curves" (see Figure 9c). The simplification was used in formulating sub-system policies and is consequently contained in the DAR and SVR systems policies.

(b) true weighting of systems objectives was replaced by preselected priorities (this is in accord with the practice embodied in official COE policies, however).

(c) using the official COE policy as standard for measuring improvements obtainable by DAR or SVR systems policies is a simplication; actual COE operations tend to be more complex and adaptable than the guide curve rules; research into that aspect is still under way.

4. *SUMMARY OF PWRRC REPORT #33*

A. CONTENTS

The report #33 contains:

(a) simulation results for the Upper Wabash River and Reservoir system;

most results were obtained for basin-wide, single storm (Gamma Function) runoff hydrographs, representing run-off depth d_k = 1", 2", ... 14";

(b) a comparison of results obtained by using three reservoir operation policy models. These are: (i) Corps of Engineers stated operating policies involving individual rule or guide curves, and control stations that introduce some element of systems operation; (ii) a systems balancing policy DAR based on reserving flood storage in the system's reservoirs in proportion to the individually controlled watershed acres; (iii) a systems balancing policy SVR based on reserving an equal percentage of flood storage in each of the system's reservoirs;

(c) generalized summaries of obtained results using the synthetic hydrologic inputs mentioned in (a);

(d) simulation results obtained when using selected historical hydrologic inputs. Results were obtained for each of the three operating policies named under (b).

B. <u>RESULTS</u>

(a) The DAR and SVR systems policies produced more flood peak reduction than the stated COE operating policy for basin wide runoff depths between 1" and 8". This includes all cases of practical interest. When using as inputs gamma-type runoff hydrographs corresponding to depths larger than 8", the COE policy performed better. Such runoff volumes are hardly realistic, however;

(b) There was little difference in the effectiveness of the DAR and SVR policies;

(c) The degree of flood peak reduction RR [%] depends not only on the adopted operating policy. The study yielded generalized information indicating how the amount of obtainable flood peak reudction depends on A_{rc} [%] = the percentage of the watershed area at section c controlled by reservoir r, and on the local flood $\bar{Q}_{cf}$ = flow rate at station c selected as the onset of flooding. For a given policy the flood peak reduction RR roughly equaled A_{rc}. Lowering of $\bar{Q}_{cf}$ makes A_{rc} less and less effective.

(d) The build software system can be useful as in answering planning and general information questions. Examples include: (i) the effect of adding or subtracting reservoirs; (ii) the effect of changing designated flood flow levels $\bar{Q}_{cf}$ at one or more locations on flood peak reduction elsewhere; (iii) the effect of reservoirs on changing the 7-day, 10-year low flow characteristics.

C. <u>LIMITATIONS AND CRITIQUE</u>

(a) The major limitation of the model is the inability to accept precipitation forecasts to generate inflow hydrographs.

(b) The usefulness of the simulation system is abridged by the lack of engineering-economic evaluation models.

5. *SUMMARY OF PWRRC REPORT #49a*

A. <u>CONTENTS</u>

The report #49a contains:

(a) a report on program revision to make the simulation model programs more modular and to decrease processing time and needed core storage.

(b) a feasibility study for geographically extending the system down to the Ohio River.

(c) the documentation of all developed sub-routines and of the main program structures for the three operating policies used;

(d) a discussion of field use of the simulation model and of the addition of forecasting;

(e) some further work on the sensitivity of time-to-peak and of peak-flood-flow to variation in the routing coefficients.

B. <u>RESULTS</u>

(a) Following program revisions the 60 seconds computation time for each COE policy run was decreased 50%. The reduction for the DAR and the SVR policy runs was 20%.

(b) An extension down to the Wabash River would increase the number of river reaches from 6 to 100 and possibly lead to an excessive growth in nested DO-loops.

(c) Field use of the model for actual daily management purposes is not feasible without adaptation by Agency Staffs. It is, however, moderately suitable for planning purposes.

C. LIMITATIONS AND CRITIQUE

(a) While efforts were made to improve main program efficiency no corresponding work was done to improve the data base management.

(b) The routines for graphic output were not yet documented.

6. *SUMMARY OF PWRRC REPORT #48*

A. CONTENTS

The report #48 records the only effort at economic evaluation of the benefits of the Upper Wabash reservoir system. Some work was done to evaluate flood damage reduction benefits near Terre Haute. This work was terminated because available time and funds did not permit the simulation model to be built out that far down stream. Thus report #48 is limited to evaluation of the water-based recreation benefit. It contains:

(a) the statement that the public controversy over the Wildcat Creek Reservoir[1)] centers on the type of recreational use of the Wildcat Creek Valley near Lafayette. Landowners and hikers oppose the uses desired by boaters and swimmers. For either type of use an estimate of expected visitation is needed in discussing project benefits.

(b) a second reservoir of the five authorized, namely Big Pine reservoir,[1)] is also controversial. An estimate of the number of expected water based recreationists per year is an important element in its evaluation;

(c) a network model for estimating the number of annual visitors that can be expected at any combination of reservoirs in the Upper Wabash Basin.

[1)] *Wildcat Creek Reservoir (also called Lafayette Lake) and Big Pine Reservoir are presently (June 1976) in the process of being de-authorized following public controversy.*

(d) the model is based on national as well as Indiana data on reservoir visitation. It reveals the dependencies of such visitation on the number of population centre within driving distance, on the size of those centra, on the driving distance, and on the attractiveness of the reservoirs. The last two factors also hold in modified form for tourists passing through the region.

(e) in addition the model incorporated the histories of capital investments in visitation facilities for each of the reservoir sites;

(f) furthermore, available estimates of local population growth and of the current and expected distributions of purchasing power were used to build the model;

(h) finally the air distances between reservoirs and population centra, often used in estimating visitor-days, were replaced by accurate estimates of actual road travel times.

B. <u>RESULTS</u>

(a) According to the current model Wildcat Creek reservoir would draw $\sum_i U_{ij}$ = 6,000,000 visitor-days in the year 2020 (i = city index, j = reservoir index).

(b) By contrast Big Pine reservoir would draw only 180,000 visitor-days in 2020.

C. <u>LIMITATIONS AND CRITIQUE</u>

(a) Ever present data limitations in studies of this type force one to use fitting curves with a small number of to be calibrated parameters; the hyperbolic dependence of visitation U_{ij} (from city i to reservoir j) on the travel time T_{ij} between the two locations, leads to overestimating the visitation U_{ij} for very small T_{ij}. This contributes to the very large U_{ij}-value found for Wildcat Creek Reservoir.

(b) All other fitting curves require more data, particularly for situations where reservoir sites are located very close or even in metropolitan areas. Since most reservoirs are sited in rural areas such data are hardly available. More general studies of the role of water areas in creating metropolitan amanities and high quality urban arrangements are needed.

III. PROJECT COMPONENTS IN NEED OF AMENDMENT

Project components that need to be amended or revised from a technical point of view will now be outlined. The reasons for such revisions will also be stated. Whether or not the necessary investments are exceeded by the expected benefits is a question discussed in the subsequent section IV.

1. *DATA BASE ORGANIZATION AND ACCESS*

The components of the system were initially constructed while drawing on a data base that could only be accessed year by year. This led to a need for considerable core storage. Currently the data base, stored in binary form on permanent file disks, can be accessed day by day, month by month, or year by year. Yet, data base formating, data retrieval and editing were never optimized after the simulation model was built and used once.

Improved (though not yet operational) revisions of data base access routines were prepared and the programs better commented. By and large the documentation given in PWRRC Report 49a is still valid excepting page 49 and 50; these two pages are obsolete.

Needed as yet were a better data base management methodology. This would include logs from original data references through current status. The primary need, however, is a more efficient structure of the data base. Secondly, a Data Base User's Manual is needed. Finally, all data should be transferred from cards to tape and procedures for regular re-recording to keep tapes in operating order should be instituted. Chapters VI and VII of this report relate the work that was accomplished to amend earlier efforts.

2. *ROUTING MODELS*

The routing model that was used is the Muskingum model. Normally the two free coefficients of this model, say C_0 and C_1, are fitted using a number of "representative" large floods. An innovation early during the project was to use all stream flow data and to employ formal optimization to obtain the best possible estimates of C_0 and C_1, namely C_0* and C_1*, respectively.

It was found that: (a) the routines for obtaining C_0* and C_1* were laborious; and (b) the model could not fit adequately both high and low flow conditions. By sub-dividing data into high and low flows and into rising and falling hydrograph portions, the overall error was reduced from 15-20% to 7-10% and short period errors from 100% to 50%. The whole procedure became yet more laborious, however. Furthermore, the coefficients did not exhibit on interpretable dependence on low or high flows or falling or rising flow rates when comparing successive reaches. Finally, the flows estimated by the model were overly sensitive to computed C_0* and C_1* values.

Progress in modeling and parameter optimization achieved in the meantime permits using more general models. These, furthermore, make it possible to use routing models that correlate reach outflow not only with reach inflow but also with precipitation data. As a result many of the hydrologic restrictions in modeling side inflow, can be eliminated. Furthermore, there is far greater freedom in selecting model structure.

A desirable revision is, therefore, to change the routing components of the model. This can be done by using more general models and more efficient parameter estimation routines. This revision is, furthermore, needed if the total simulation model is ever to be used in a "predictive mode" i.e. accepting precipitation forecasts; this is essential for daily management purposes. The revision is also essential if the Upper Wabash model is to be extended geographically.

3. *OPERATING POLICY ROUTINES*

Since the operating policy routines were written, additional research has been done on the daily operating procedures used by the Corps of Engineers. This work permits improvements in the COE policy routine of the Upper Wabash model.

Secondly, the DAR and SVR systems policies contain a large number of DO-loops. Their number may become excessive when the Wabash model is extended downstream. Some revision of the iteration procedure for satisfying constraints at stage control points in the basin, is likely to be needed.

4. *EXTENDING THE SIMULATION MODEL FURTHER DOWNSTREAM*

The PWRRC Report #49a contains in Chapter VIII preliminary work on three types of extensions of the Wabash Simulation Models. These are:

(a) an increase in modularity and efficiency of the model routines;

(b) the "addition" of a forecasting mode of operation;

(c) the geographic extension from Covington down to the Ohio River.

The extensions (a) and (b) have been discussed above. The extension (c) first of all requires (a) and (b). All needed stream flow data are already part of the current data base except for some Illinois stations giving tributary flows. Only part of the needed precipitation data is available in the data base. A special effort to complement these and, in particular, to compute missing data, is needed.

IV. RECOMMENDED SURFACE WATER SIMULATION WORK

Maintaining the current Upper Wabash simulation software is worthwhile only if benefits of further use exceed the costs. However, making this comparison for research products is very difficult. The benefits can be divided roughly into three groups:

A - Training and education benefits:
- (a) Student Education
- (b) Educational research program continuity

B - Advancing the state of the art:
- (a) Scientific
- (b) Engineering Management

C - Yielding results for direct use:
- (a) by operations staff of agencies
- (b) by planning staff of agencies
- (c) by general public

1. *BENEFITS OF FURTHER WORK*

A. The probability of benefits in group A is generally good. For A-a benifits of the Upper Wabash work one may point to its carry-over into the acclaimed work done by Dr. T. P. Chang in river quality simulation. Dr. Chang gained national recognition as head of the computation branch of the Indiana Stream Pollution Control Board.(ISPLB).

The benefits of continuity, phase A-b, are significant because simulation modeling is the only type of model feasible whenever larger daily water management problems are to be studied and solved. Because the models are large and complex, interruption can be especially damaging if one does not perform a minimum of maintenance of the data base and program library.

B. The Upper Wabash modeling created operating policy routines. These solved some and brought into clear view other difficulties of operational control and the modeling thereof. While the developed iterative methods need revision for larger systems, they are useful to systems having 2-5 flood control reservoirs.

An important contribution, namely multi-state variable control of reservoir operation, requires further work to become fully developed.

C. Only a small fraction of academic research leads to an interest by or to results of immediate usefulness to agency staffs responsible for day to day water management. In this respect the Upper Wabash modeling work has at least scored. It was presented on a number of occasions to agency staffs. It has led directly to funded research by an operations branch of the Corps of Engineers (for simulation models, Green River Basin, Kentucky). Similar interest by basin planning staffs concerned with Indiana can be expected provided the model can be expanded to encompass a geographically larger system so that downstream flood control benefits can be evaluated.

Following such extension, the model is of interest to EPA and the ISPCB.

While obtainable model results relate to a key point (downstream flood control) of the present Wildcat Creek Reservoir controversy, its use by the public was unpredictable from the start. On four occasions public contact about the model occurred. Once the principle investigator brought it to the attention of the Lafayette Radio station audience. It turned out that subjective information and interpretations can become valued more than "objective" model information. Currently, the de-authorization procedures for the Wildcat Reservoir are underway so that particular study motivation is being eliminated.

In summary it may be said that an analysis of project results indicates that expected benefits of further simulation work within reach are weighted towards tasks for which universities are most suited, namely professional education and research.

2. *RECOMMENDATIONS*

It is recommended that:

(a) the routing components of the Upper Wabash simulation model be replaced by multi-parameter models that incorporate precipitation data;

(b) the operating policy routines be revised to permit geographic extension of the system;

(c) the model be enlarged to cover Wabash reaches down stream of Covington;

(d) the flood control effects be determined of adding or not adding the Wildcat Creek Reservoir to the system in order to be able to later analyze the controversy as a case study;

(e) the 7-day, 10-year low flow statistic (which is basic to all PL208 studies) used by EPA and the ISPCB be revised to reflect the presence of new reservoirs in the Wabash Basin; the simulation model is the only way to do this with reasonable accuracy;

(f) the simulation model be adapted for studying the relative quality effects of point and non-point pollution.

The first of these recommendations has been implemented. The work is reported in Chapter V of this report. The other recommendations cannot now be given further consideration for lack of funding.

V. MULTI-INPUT LINEAR ROUTING MODEL AND APPLICATION

1. *INTRODUCTION*

In sections II-2-C and III-3 the method for determining the Muskingum routing model coefficients was critiqued. The reason was its laboriousness. Furthermore, the Wabash simulation model could not be used in a predictive mode since the model did not accept precipitation input. Additional work was done to improve the Wabash simulation model in two ways. First a more general routing model, the multi-input linear routing model, has been formulated and used. Secondly, in the development of that model, precipitation data were used. These make it possible to have precipitation predictions among the Wabash model inputs. The section V reports in this work.

The multi-input linear routing model is proposed as a replacement of the Muskingum routing model component of the Upper Wabash Surface Water Simulation Model. The multi-input linear routing model (MILRM) is a generalized linear hydrologic routing model. It enables one to utilize directly all relevant data series, including at the same time all tributary inflows and precipitation data to estimate contributions of ungaged side inflow. There is no need to account for such data as part of the reach inflow or outflow. Therefore, it is simpler than the Muskingum models proposed in Report #27 (Toebes and Chang, 1972) while retaining all advantages of a linear model such as the Muskingum model.

2. *THEORETICAL BACKGROUND*

MILRM is a black box type model which relates the output to each of several inputs. Consider Fig. V-1 portraying a channel reach having: a gaged inflow I(t); a gaged outflow O(t); gaged tributary inflows $T_i(t)$, i=1,...s. Suppose further that local average rainfall data, P(t), falling onto ungaged side inflow contributing area A, is available. A black box representation of Fig. V-1 is shown in Figure V-2.

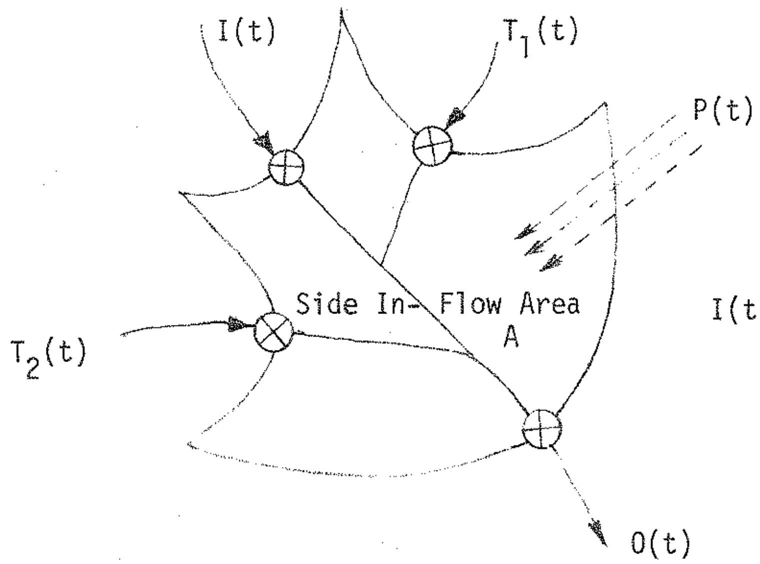


FIGURE V-1 - CHANNEL REACH

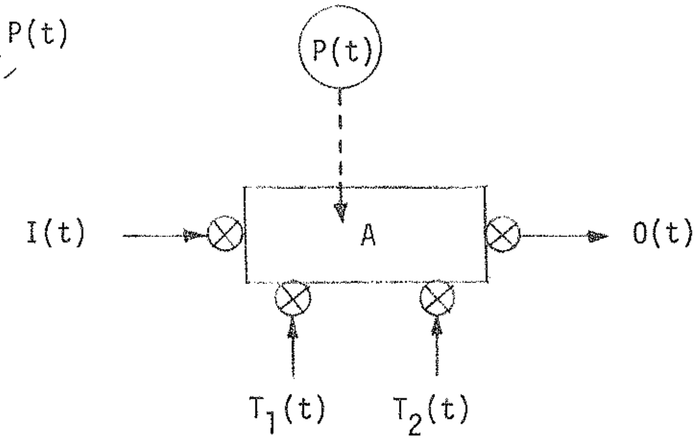


FIGURE V-2 - BLACK BOX MODEL REPRESENTATION

The system is assumed to be linear, stationary and deterministic. The generalized multi-input linear routing model M(l,m,n,q) is expressed by the equation:

$$O(t) = \sum_{\tau=1}^{\ell} CO(\tau)\cdot O(t-\tau) + \sum_{\tau=0}^{m} CI(\tau)\, I(t-\tau)$$
$$+ \sum_{\tau=0}^{n} CP(\tau)\cdot C\cdot A\cdot P(t-\tau)$$
$$+ \sum_{i=1}^{S} \sum_{\tau=0}^{q_i} CT_i(\tau)\cdot T_i(t-\tau) \qquad \text{(V-1)}$$

where: ℓ, m, n, q_i = orders of model components

S = no. of gaged tributary inflows

$CO(\cdot)$, $CI(\cdot)$, $CP(\cdot)$, $CT_i(\cdot)$ = linear model parameters

C = conversion factor

A = ungaged side inflow contributing area

Note that the Muskingum model is a special case of Eq. V-1, namely that of only one input, I(t), and ℓ=m=1 so that:

$$O(t) = CO(1)\ O(t-1) + CI(0)\ I(t) + CI(1)\ I(t-1).$$

Another special case is the unit hydrograph model. For that case there is only one input P(t) so that the equation becomes:

$$O(t) = \sum_{\tau=0}^{n} CP(\tau)\cdot C\cdot A\cdot P(t-\tau)$$

It is interesting to note that, by substituting $O(t-\tau)$ terms in Eq. V-1 using the Eq. V-1, that same equation can be rearranged into a form identical to the one representing a multi-input linear system model commonly used in control theory

$$O(t) = \sum_{\tau=0}^{\infty} CI'(\tau)\ I(t-\tau) + \sum_{\tau=0}^{\infty} CP'(\tau)\cdot C\cdot A\cdot P(t-\tau) + \sum_{i=1}^{S}\sum_{\tau=0}^{\infty} CT_i'(\tau)\ T_i(t-\tau) \qquad \text{(V-2)}$$

where: $CI'(\cdot)$, $CP'(\cdot)$, $CT_i'(\cdot)$ = linear kernel or response functions, which can be expressed in terms of linear model parameters and model orders. For example: $CI'(\tau)$, $\tau = 0,1,\ldots,\infty$ can be expressed by a function:

$$f\{CO(\tau_\ell),\ CI(\tau_m)\};\ \tau_\ell = 1,2,\ldots,\ell;\ \tau_m = 0,1,\ldots,m.$$

Similarly for $CP'(\tau)$, $\tau = 0,1,\ldots,\infty$:

$$g\{CO(\tau_\ell),\ CP(\tau_p)\};\ \tau_\ell = 1,2,\ldots,\ell;\ \tau_p = 0,1,\ldots,n \text{ etc.}$$

3. *MODEL COMPUTATION*

The model expressed by equation V-1 has several terms from the same data series. For example: I(t), I(t-1),.. $I(t-\tau)$ all represent identical series I(t) except for having time lags. If the equation V-1 is used directly in digital computation, the required amount of central memory of the computer is enormous. For example, assume a model having only 3 inputs; I(t), P(t) and T(t), and the use of a model order of 1 for all series. Further, assume that all input series consists of 10 years of daily data. The amount of central memory required would then be in excess of 87600 60-bit words. This amount is too great for most medium size computers.

However, the problem can be resolved by noting that the equation V-1 can be rewritten in a more compact form as follows:

$$O(t) = \sum_{m=1}^{M(1)} H_1(m)\ D_1(t-m) + \sum_{n=2}^{N}\sum_{m=0}^{M(n)} H_n(m)\ D_n(t-m) \qquad \text{(V-3)}$$

where: $H_1(m),\ D_1(t) \equiv CO(\tau),\ O(t)$ respectively
$H_2(m),\ D_2(t) \equiv CI(\tau),\ I(t)$
$H_3(m),\ D_3(t) \equiv CP(\tau),\ C\cdot A\cdot P(t)$
$H_4(m),\ D_4(t) \equiv CT_1(\tau),\ T_1(t)$

$H_N(m),\ D_N(t) \equiv CT_s(\tau),\ T_s(t)$

N = no. of data series inputs and output series = $S + 3$
$M(n)$ = model order for component n.

Furthermore, in equation V-3, the linear model parameters $H_1(m)$ and $H_n(m)$ may not be significant for all values of $m=0,1,2,\ldots,M(n)$. If not, and assuming that the set of relevant m-values is continuous, one may adopt a lower index $MS(n)$ and an upper index $MR(n)$ representing the starting order and the ending order. In that case the equation V-3 becomes

$$O(t) = \sum_{n=1}^{N} \sum_{m=MS(n)}^{MR(n)} H_n(m)\ D_n(t-m) \qquad \text{(V-4)}$$

The equation V-4 will be symbolically abbreviated to $M\left\langle\frac{MS(n)}{MR(n)}\right\rangle$. By way of illustration, consider a two-input model, with $I(t)$ and $P(t)$ as inputs. Then the equation:

$$\begin{aligned} O(t) = {} & CO(1)\ O(t-1) + CI(0)\ I(t) + CI(1)\ I(t-1) \\ & + CP(0)\cdot C\cdot A\cdot P(t) + CP(1)\cdot C\cdot A\cdot P(t-1) \\ & + CP(2)\cdot C\cdot A\cdot P(t-2) \end{aligned} \qquad \text{(V-5a)}$$

for example, may be abbreviated to:

$$M\left(\frac{1\ 0\ 0}{1\ 1\ 2}\right) \qquad \text{(V-5b)}$$

Other illustrations of the use of Eq. V-5 are found in the Table V-1. The equation V-4 can still not be used in the reduction of required storage. If we let:

Y_t = reach outflow on day t = $O(t) = D_1(t)$;
$X_t(j)$ = array containing model components values for a given day t;
$\Theta(j)$ = array of linear model parameters corresponding to the $X_t(j)$;
MX = no. of model components

the equation V-4 can be reduced to:

$$Y_t = \sum_{j=1}^{MX} \Theta(j)\ X_t(j) \qquad \text{(V-6)}$$

where:

$$MX = \sum_{n=1}^{N} \{MR(n) - MS(n) + 1\} = \text{total number of terms of the model.}$$

$$\begin{aligned} X_t(j) &= D_1(t-\ell) \\ \Theta(j) &= H_1(\ell) \end{aligned} \qquad \text{in which} \left\{ \begin{aligned} \ell &= MS(1),\ MS(1)+1, \ldots, MR(1) \\ j &= 1,2,\ldots,J1 \\ J1 &= MR(1) - MS(1) + 1 \end{aligned} \right.$$

$$\begin{aligned} X_t(j) &= D_2(t-\ell) \\ \Theta(j) &= H_2(\ell) \end{aligned} \qquad \text{in which} \left\{ \begin{aligned} \ell &= MS(2),\ MS(2)+1, \ldots, MR(2) \\ j &= J1+1,\ J1+2,\ldots,J2 \\ J2 &= J1 + \{MR(2) - MS(2) + 1\} \end{aligned} \right.$$

$$\begin{aligned} X_t(j) &= D_3(t-\ell) \\ \Theta(j) &= H_3(\ell) \end{aligned} \qquad \text{in which} \left\{ \begin{aligned} \ell &= MS(3),\ MS(3)+1, \ldots, MR(3) \\ j &= J2+1,\ J2+2,\ldots,J3 \\ J3 &= J2 + \{MR(3) - MS(3) + 1\} \end{aligned} \right.$$

and so on.

The equation V-6 is the desired condensed form of the equation V-4. The Eq. V-6 is now used to estimate Y_t and also to estimate $\Theta(j)$ using an iterative procedure as discussed in the section V-4,5. That procedure will permit any size of model to be evaluated by computer. The equations V-1,4 display best the structure of the model whereas the equation V-6 is a computation-oriented representation.

4. *PARAMETER ESTIMATION*

Before one can use Eq. V-6, the limit MX must be known. Therefore, the model orders MS(n) and MR(n) must be selected prior to the estimation of parameters. The criterion for selecting model orders MS(n) and MR(n) will be discussed in the section V-5. Suppose for the moment that the model orders have been selected. Then the parameters can be estimated by the method of least squares which amounts to:

$$\min[E] = \min_{\Theta(j)} \left[\sum_{t=1}^{T} \{Y_t - \hat{Y}_t\}^2 \right] \tag{V-7}$$

where: $Y_t = O(t) = D_1(t)$ = historical outflow

$\hat{Y}_t = \hat{O}(t)$ = predicted outflow using Eq. V-6:

$$\hat{Y}_t = \sum_{j=1}^{MX} \Theta(j)\, X_t(j) \tag{V-8}$$

The quantity E in Eq. V-7 is minimized by taking partial derivatives and equating them to zero:

$$\frac{\partial E}{\partial \Theta(j)} = 0 \qquad \text{for all } j=1,\ldots,MX \tag{V-9}$$

It can be shown that the equation V-7, V-8, and V-9 yield the following result:

$$\hat{\underset{\sim}{\Theta}} = A \cdot B \tag{V-10}$$

where: $A = \left[\sum_{t=1}^{T} \underset{\sim}{X}_t^T \underset{\sim}{X}_t\right]^{-1}$; $B = \left[\sum_{t=1}^{T} \underset{\sim}{X}_t^T Y_t\right]$

$\hat{\underset{\sim}{\Theta}}$ = estimated parameter vector = $[\Theta(1), \Theta(2), \ldots, \Theta(MX)]^T$

$\underset{\sim}{X}_t$ = vectors of model component for day t

$$\underset{\sim}{X}_t = \begin{bmatrix} X_t(1) \\ X_t(2) \\ \vdots \\ X_t(MX) \end{bmatrix}$$

The regression coefficient R^2 is usually taken as an indication of model goodness. R^2 is estimated from the error.

$$Y_t - \hat{Y}_t = \text{error} \tag{V-11}$$

$$\text{Sum of square error} = \sum_{t=1}^{T} \{Y_t - \hat{Y}_t\}^2 = E \tag{V-12}$$

$$\text{Sum of squares of data} = \sum_{y=1}^{T} \{Y_t\}^2 = SS \tag{V-13}$$

$$R^2 = 1 - E/SS \tag{V-14}$$

The f-statistics for the model and for each individual parameters are obtained from:

$$f_{model} = \left(\frac{T-MX}{MX}\right)\left(\frac{R^2}{1-R^2}\right) \tag{V-15}$$

$$f(j) = \frac{\{\Theta(j)\}^2}{Var\{\Theta(j)\}} \tag{V-16}$$

where: $Var\{\Theta(j)\} = (a_{jj})\left(\frac{E}{T-MX}\right)$;

(a_{jj}) = diagonal component of A in Eq. V-10.

5. *MODEL ORDER SELECTION*

As mentioned in section V-4 the model orders MS(n) and MR(n) must be selected first using some criterion such as the R^2-gain criterion, the f-statistic criterion, and the whiteness of the residue criterion. A strict use of any single one of these criteria did not provide for a satisfactory procedure. A procedure for estimating MS(n) and MR(n) was developed that involved the R^2-gain criterion and the use of f-statistic values.

The basic concept of the R^2-gain criterion is to choose the model orders MS(n) and MR(n) such that the R^2 obtained is greater than a minimum specified value and that the gain in R^2, i.e., ΔR^2, obtained by increasing the model order of any series is less than a second specified value.

In the computations, the following values of model orders are first assumed:

$$MS(1) = 1 \text{ and other } MS(n) = 0 \text{ for } n \geqslant 2$$

$$MR(n) = m, \; n = 1,2,\ldots,N \qquad \text{(V-17)}$$

Selected are the minimum value R^2 = 97.5% and the critical gain value $(\Delta R^2)_{min}$ = 0.25%. The value of m is incremented by one each time until $(R_m^2 - R_{m-1}^2) \leq (\Delta R^2)_{min}$ and $R^2 \geq R_{min}^2$. Then the model order (m-1) is tentatively chosen and the final choice of MS(n) and MR(n) are then chosen by considering the f-statistics (Equation V-16). The parameters that give values of f-statistics that are too small are deleted from the model. The values of MS(n) and MR(n) then are specified and hence used to estimate the final parameters. The value of R^2 will then be compared with that obtained previously. If the new R^2 is less than R_m^2 by more than $(\Delta R^2)_{min}$, then MS(n) and MR(n) are adjusted with the help of the f-statistics of each parameter until the condition $(R_m^2 - R_{revised}^2) \leq (\Delta R^2)_{min}$ is satisfied.

The Figure V-3 shows in a compact but clear diagram the MILRM parameter estimation algorithm. The actual lisitng is found in Appendix D.

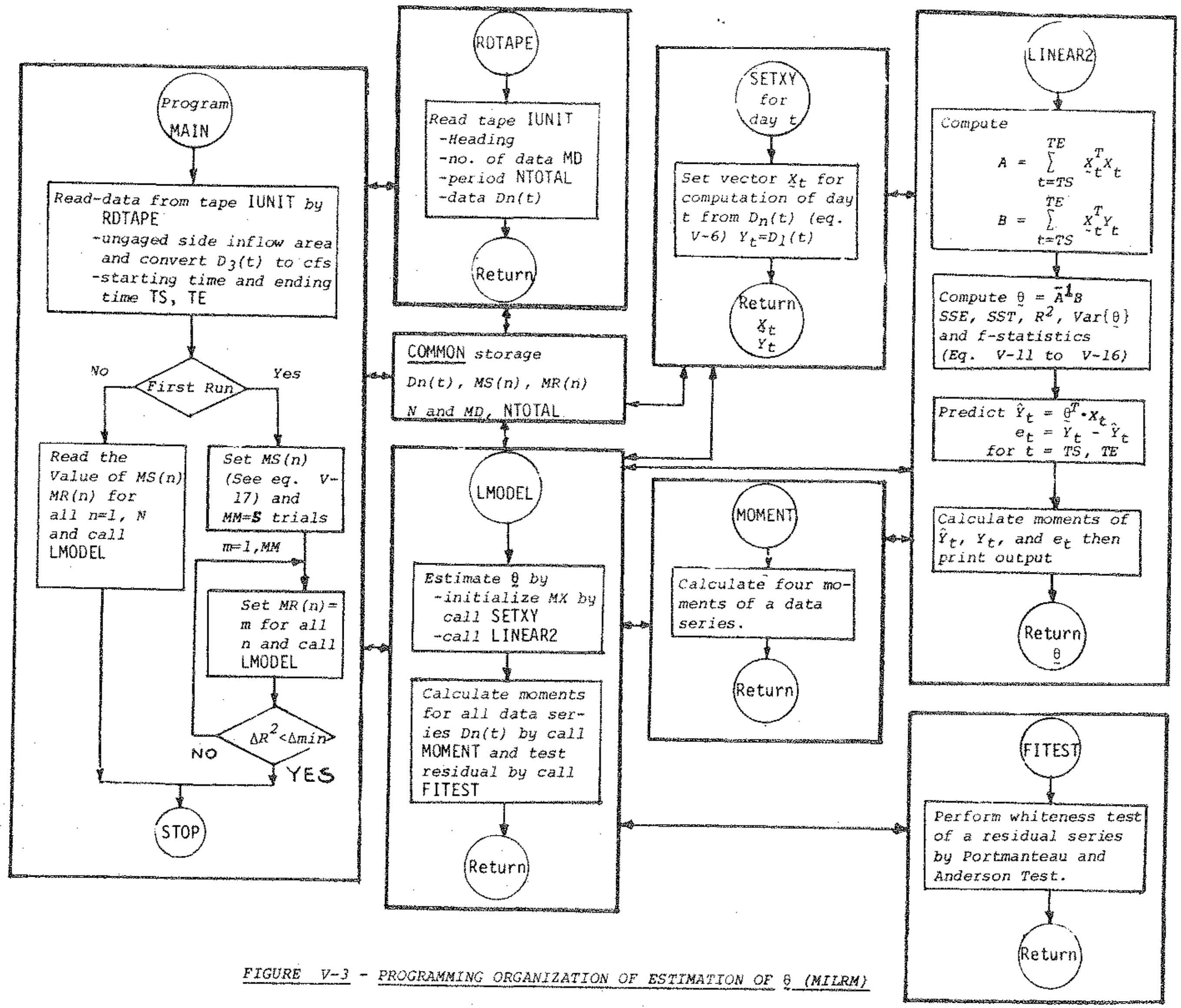


FIGURE V-3 - PROGRAMMING ORGANIZATION OF ESTIMATION OF θ (MILRM)

6. *MODES OF MODEL USE*

There are two modes of model use, namely the forecasting mode and the response mode.

Forecasting Mode

MILRM-models can be used in planning as well as the real-time operation of water resource systems. In operation of large surface water systems, forecasted data will be used. The MILRM models are suitable building blocks for implementing operating policies involving reach outflow forecasts.

Consider first a single reach component of a larger simulation model for any single reach represented by an MILRM model, the "L-day ahead" forecast of reach outflow, to be designated by $\hat{Y}_{t+L|t}$, can be obtained from:

$$\hat{Y}_{t+L|t} = \sum_{j=1}^{MX} \theta(j)\hat{X}_{t+L|t}(j) \tag{V-18}$$

representing the computational equation for model output, or alternatively by:

$$\hat{Y}_{t+L|t} = \sum_{n=1}^{N} \sum_{\tau=MS(n)}^{MR(n)} H_n(\tau)\hat{D}_n(t+L-\tau|t) \tag{V-19}$$

representing the more physical form of the model for computing reach outflow.

The subscript $t+L|t$ signifies that the $\hat{Y}$-estimate and the vector $\hat{X}$ are based on measured values up to day t and on exogeneously forecasted values for day t+1, t+2, ..., t+L. Analogously the notation $\hat{D}_n(t+L-\tau|t)$ signifies that for $t+L-\tau > t$ that $\hat{D}_n$ are forecasted values, whereas for $t+L-\tau \le t$ the $\hat{D}_n = D_n$ are historic values.

By way of graphic illustration, the simulation model when used in a forecasting mode utilizes input and produces output as follows:

Input		Model		Output
$D_n^h(t)$, $n \neq 1$; $\bar{Y}_t$ $D_n^h(r)$; $r<t$ - - - - - - - - - - $\hat{D}_n(f\|t)$; $t+1 \le f \le t+L$	$\longrightarrow$	SIMULATION MODEL	$\longrightarrow$	$\hat{Y}_{f\|t} = \hat{O}(f\|t)$; $t+1 \le f \le t+L$

$D_n^h(t)$, $\hat{Y}_t$, $D_n^h(r)$ - represents historic or measured data vectors.

$\hat{D}_n(f|t)$ - represents a series of L data (for each input vector component) obtained from forecast generated exogeneously from the simulation model.

FIGURE V-4 - USE OF SIMULATION MODEL IN FORECASTING MODE. THIS USE ARISES IN FORECAST ABILITY STUDIES OF THE MODEL (IN WHICH CASE HISTORIC DATA SERIES ARE USED FOR D_n*) AND IN REAL TIME OPERATION USE OF THE MODEL (IN WHICH CASE MEASURED AND FORECASTED DATA ARE USED FOR* D_n*).*

Response Mode[1)]

In planning studies, simulation models are commonly used in the response mode. Again consider first a single reach component of a larger simulation model. For a single reach represented by an MILRM model, the day t outflow $\hat{O}(t)$ or $\hat{Y}_t$, can be obtained from:

$$\hat{Y}_t = \sum_{j=1}^{MX} \theta(j)\hat{X}_t(j) \tag{V-20}$$

representing the computational equation for model output, or, alternatively, by:

$$\hat{Y}_t = \sum_{n=1}^{N} \sum_{\tau=MS(n)}^{MR(n)} H_n(\tau)\hat{D}_n(t-\tau) \tag{v-21}$$

where: $\hat{D}_1(t) = \hat{Y}(t)$ for $t > 0$
$\hat{D}_n(t-\tau) = D_n(t-\tau)$ for $t-\tau \le 0$

---

1) an alternative designation could be "Generating Mode" for this section concerns the use of the model as a data series generator, i.e. an algorithm that generates synthetic output data series from synthetic input data series

By way of graphic illustration the simulation model when used in a response mode utilizes input and produces output as follows:

$$\boxed{\begin{array}{l} D_n(t);\ n\neq 1 \\ D_n(r);\ n\neq 1,\ r<t \\ \hline \hat{Y}_r \quad ;\ r<t \end{array}} \longrightarrow \boxed{\begin{array}{c} \text{SIMULATION} \\ \text{MODEL} \end{array}} \longrightarrow \hat{Y}_t = \hat{O}\ (t)$$

FIGURE V-5 - USE OF MODEL IN RESPONSE MODE. THIS USE ARISES IN PLANNING STUDIES. THE $D_n(t)$ AND $D_n(r)$ DATA SERIES MAY BE HISTORICAL OR SEPARATELY GENERATED DATA SERIES. THE $\hat{Y}_r$ DATA IN A SERIES GENERATED BY THE MODEL ITSELF.

Illustration

The use of an MILRM model in a forecasting mode for say an $M\{\frac{1,\ 1,\ 1}{1,\ 0,\ 0}\}$ model, involves the repeated evaluation of Eq. V-19. For the one-day and the two-day ahead forecasts, respectively, the equations take the following forms:

$$\hat{Y}_{t+1|t} = H_1(1)D_1(t) + H_2(0)\hat{D}_2(t+1|t) + H_2(1)D_2(t) + H_3(0)\hat{D}_3(t+1|t) + H_3(1)D_3(t)$$

$$\hat{Y}_{t+2|t} = H_1(1)\hat{D}_1(t+1|t) + H_2(0)\hat{D}_2(t+2|t) + H_2(1)\hat{D}_2(t+1|t) + H_3(0)\hat{D}_3(t+2\ t) + H_3(1)\hat{D}_3(t+1|t)$$

where: $\hat{D}_1(t+1|t) = \hat{Y}_{t+1|t}$

7. *ERRORS IN MODEL USE*

Forecasting Mode

When an MILRM model is used in the forecasting mode, the output error (considering a single output component) will be defined as:

$$E_f = O_f - O_f^h; \qquad f = t+1, t+2, \ldots, t+L \tag{V-22}$$

where h signifies "historic." The error accumulates over L. Furthermore, in real time operation, the forecast error and the model use error compound and accumulate over L.

In considering the "goodness of the model" when used in the forecasting mode one should separate forecast error from model error. Therefore, historic rather than forecast data enter into the computation of the error series E_f. The next sections will show the result of using the model for one day ahead forecasts.

Response Mode

When an MILRM model is used in the response mode, the output error (considering a single output component) will be defined as:

$$E_s = O_t^g - O_t^h \tag{V-23}$$

g = generated in response mode

h = historical

The error E_s accumulates over T, i.e. the total simulation period. Note that E_s is defined only if O_t^g is generated using historical data, I_t^h, as input.

8. *APPLICATION TO UPPER WABASH WATER SYSTEM*

The replacement of the Muskingum routing components in the Wabash River Simulation Model by the newly developed MILRM components will now be considered. A slightly modified systems graph of the Upper Wabash Simulation Model (described in PWRRC Report #27) is shown in Figure V-6. The simulation model is segmented into 6 reach components or MILRM sub-systems. The end sections coincide with gaging stage corss-sections in the Wabash River.

The parameters and the model orders for each MILRM sub-system are shown in the Table V-1. The model parameters were estimated using 10 years of historical streamflow as well as precipitation records (1960-1969). The local precipitation over the portion of the watershed contributing ungaged side inflow, was obtained by averaging the data of selected neighboring precipitation stations.

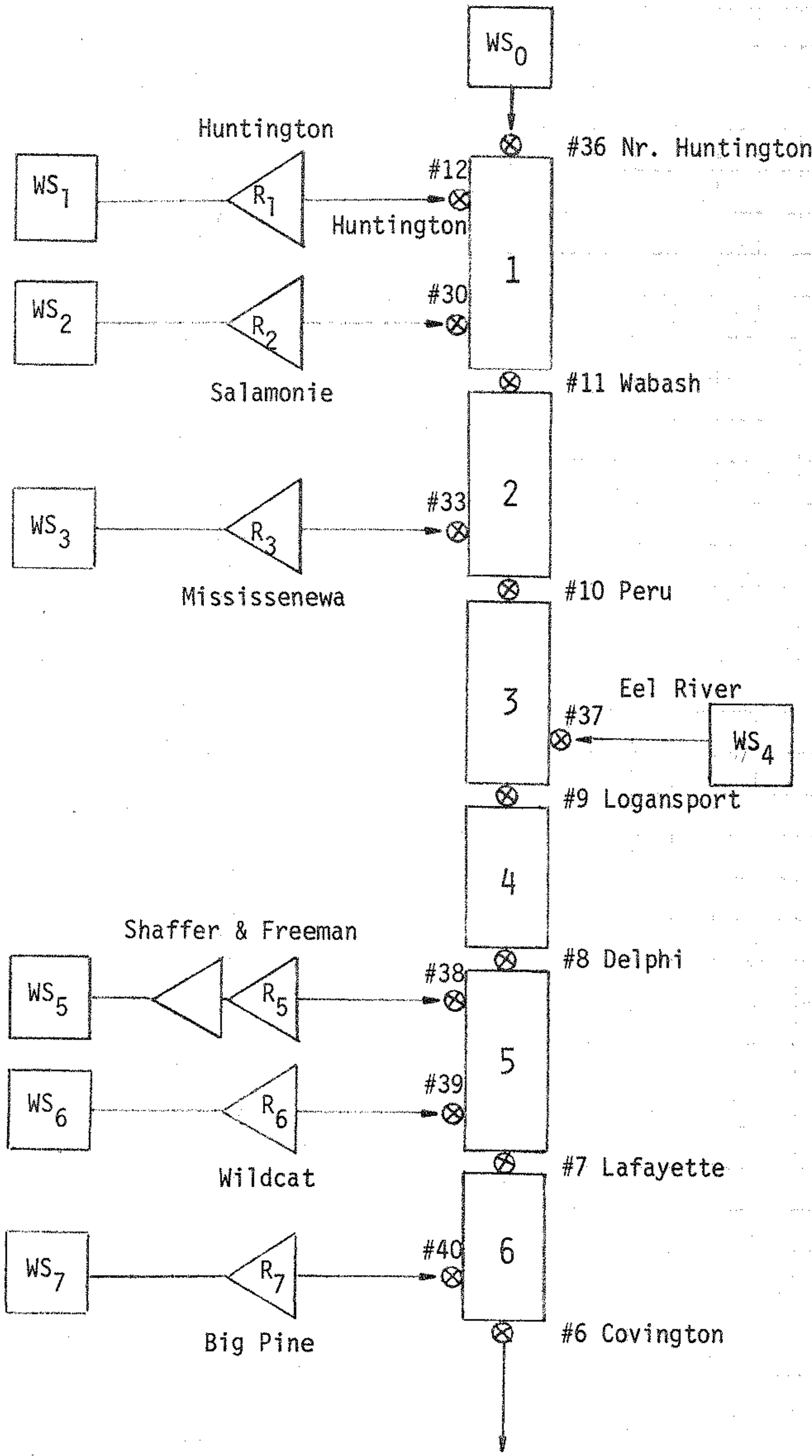


FIGURE V-6 - SYSTEMS MODELING FOR UPPER WABASH BASIN

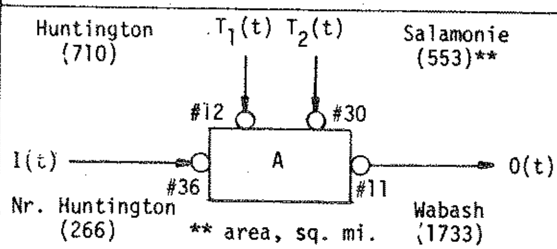
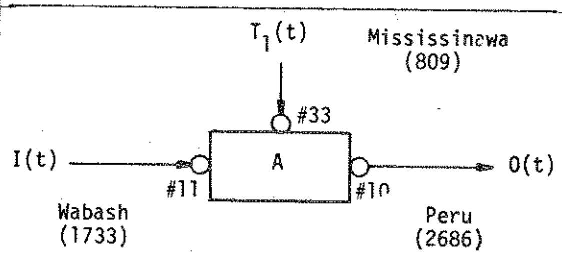
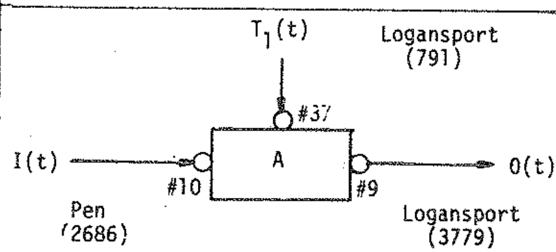
Model No.	Sub-system (Black Box Representation)	Side inflow contributing area A, local rainfall stations	Model Order $M\{\frac{MS(n)}{MR(n)}\}$, Regression Coefficient R^2	lag τ	Parameters of lag τ: $CO(\tau)$ or $H_1(\tau)$	$CI(\tau)$ or $H_2(\tau)$	$CP(\tau)$ or $H_3(\tau)$	$CT_1(\tau)$ or $H_4(\tau)$	$CT_2(\tau)$ or $H_5(\tau)$
1		A = 204 mi^2	$M\{\frac{1,0,0,0,0}{1,1,0,1,0}\}$	0		1.2779	0.03823	1.1142	0.5729
				1	0.4853	-0.5627	-	-0.5832	-
				2					
		Huntington Wabash Salamonie	R^2 = 97.95% MX = 7	3					
2		A = 144 mi^2	$M\{\frac{1,0,0,0}{1,1,0,1}\}$	0		0.8934	0.00257	0.8971	
				1	0.5688	-0.4779	-	-0.3896	
				2					
		Wabash Marion 2N	R^2 = 99.14% MX = 6	3					
3		A = 302 mi^2	$M\{\frac{1,0,0,0}{1,2,0,1}\}$	0		0.9528	0.00252	1.2091	
				1	0.5546	-0.3735	-	-0.6608	
				2		-0.0966	-		
		Logansport Columbia City Radio WSAL	R^2 = 99.54% MX = 7	3					

TABLE V-1 SUB-SYSTEM MODELS AND THEIR BLACK BOX REPRESENTATION WITH PARAMETERS

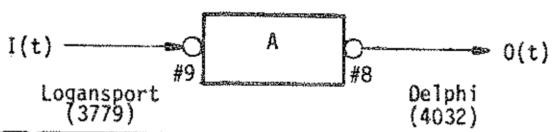
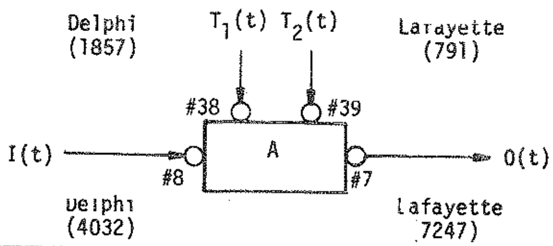
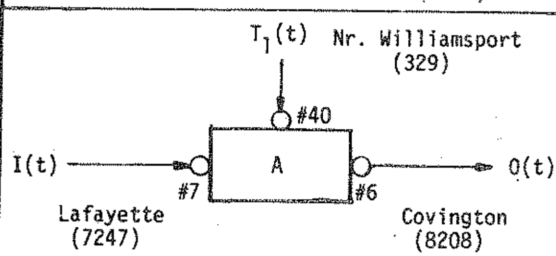
Model No.	Sub-system (Black Box Representation)	Side inflow contributing area A, local rainfall stations	Model Orders $M\{\frac{MS(n)}{MR(n)}\}$, Regression Coefficient R^2	lag τ	$CO(\tau)$ or $H_1(\tau)$	$CI(\tau)$ or $H_2(\tau)$	$CP(\tau)$ or $H_3(\tau)$	$CT_1(\tau)$ or $H_4(\tau)$	$CT_2(\tau)$ or $H_5(\tau)$
4		A = 253 mi^2 Logansport Delphi	$M\{\frac{1,0,0}{1,2,0}\}$ R^2 = 98.86% MX = 5	0		0.7155	0.05609		
				1	0.6538	-0.1654			
				2		-0.2113			
				3					
5		A = 567 mi^2 Delphi Lafayette	$M\{\frac{1,0,1,0,0}{1,2,1,2,0}\}$ R^2 = 99.50% MX = 9	0		0.6354	0.03506	0.4792	0.4316
				1	0.6736	0.0220		0.2596	
				2		-0.3113		-0.4017	
				3					
6		A = 632 mi^2 Lafayette Covington	$M\{\frac{1,0,0,0}{1,1,0,0}\}$ R^2 = 99.29% MX = 5	0		0.2820	0.3488	1.1010	
				1	0.4549	0.2935	-	-	
				2					
				3					

TABLE V-1 SUB-SYSTEM MODELS AND THEIR BLACK BOX REPRESENTATION WITH PARAMETERS (CONTINUED)

9. *COMPARISON OF MILRM AND MUSKINGUM MODEL COMPONENTS*

In the original Upper Wabash Simulation (See PWRRC Report #27) the simulation model was segmented from river junction to river junction. With reference to Fig. V-7-a the Muskingum model component provides an output QO(t) at point D given inputs QI(t) at point U. However, the MILRM model component provides an output O(t) at point B given the input I(t) at point A and the inputs $T_1(t)$ and $T_2(t)$. The outputs QO(t) and O(t) are not directly comparable.

A "Muskingum generated" output $O_m(t)$ at point B was obtained and compared to the MILRM generated O(t). Each was then compared to historical $O^h(t)$ data to judge the relative performance of the simulation model components.

The $O_m(t)$ data series were obtained in the following manner.

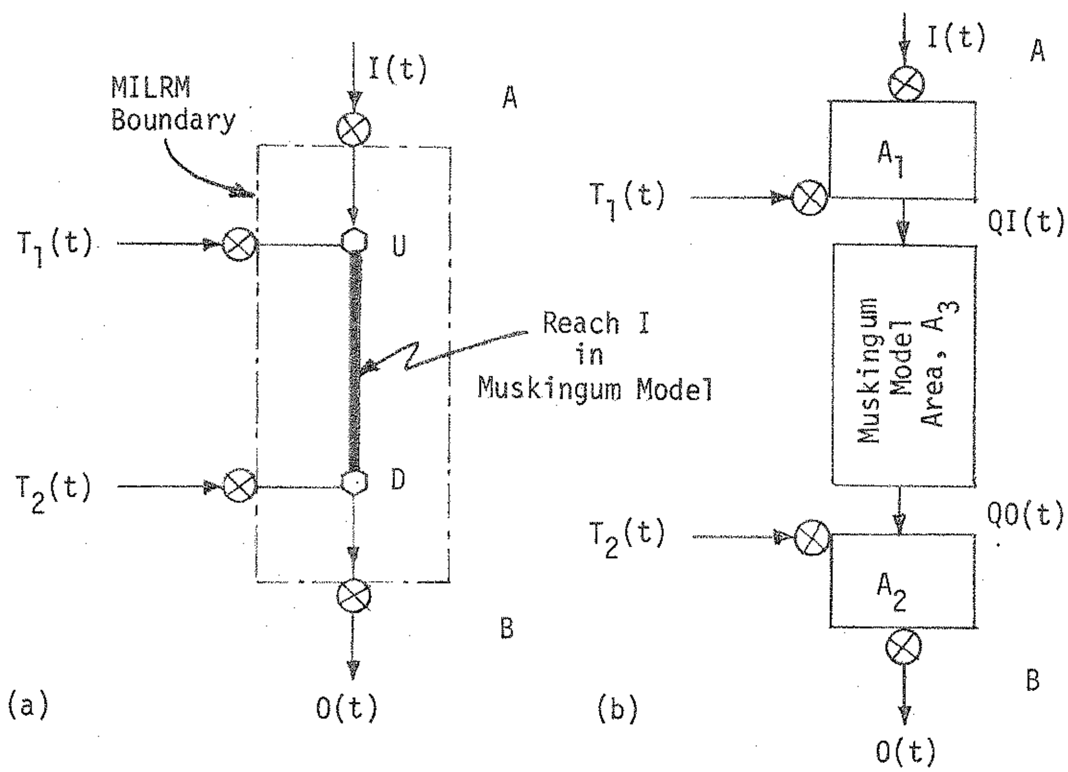


FIGURE V-7 - RIVER REACH DEFINITION FOR MUSKINGUM AND MILRM MODELS AND BLACK BOX REPRESENTATIONS

Given: $QI(t)$ = reach inflow for Muskingum model

$QO(t)$ = reach outflow for Muskingum model

A_r, $Q_r(t)$ = area and flow at the referrent station.

To find the value of QI(t) and QO(t) from historical records at gaging stations, the following equations are used.

$$QI(t) = I(t) + T_1(t) + \left(\frac{A_1 + \frac{A_3}{2}}{A_r}\right) Q_r(t)$$

$$QO(t) = O(t) - T_2(t) - \left(\frac{A_2 + \frac{A_3}{2}}{A_r}\right) Q_r(t)$$

After obtaining QI(t) and QO(t), one can then make a computation in either forecasting mode or response mode as discussed in Chapter V in order to compare the results with MILRM results in the corresponding mode. The parameters C_0, C_1, C_2 are listed in Report #27.

<u>Forecasting Mode</u> (one day ahead)

$$\hat{QO}(t) = C_0\ QO(t-1) + C_1\ QI(t-1) + C_2\ QI(t) \qquad \text{(V-21)}$$

$$\hat{O}_m(t) = \hat{QO}(t) + T_2(t) + \left(\frac{A_2 + \frac{A_3}{2}}{A_r}\right) Q_r(t) \qquad \text{(V-22)}$$

<u>Response Mode</u>

$$\hat{QO}(t) = C_0\ \hat{QO}(t-1) + C_1\ QI(t-1) + C_2\ QI(t) \qquad \text{(V-23)}$$

$$\hat{O}_m(t) = \hat{QO}(t) + T_2(t) + \left(\frac{A_2 + \frac{A_3}{2}}{A_r}\right) Q_r(t) \qquad \text{(V-24)}$$

The errors in both modes of use were discussed in section V-7.

The Muskingum and MILRM models has been used to calculate the reach outflow for each of the six reach models, shown in Figure V-6. The calculation has been made for both modes of use, for 1967 water year. The Figures V-7 and V-8 show the computed outflows, the historical outflow and the errors for reach model #1 in the response and the forecasting mode, respectively.

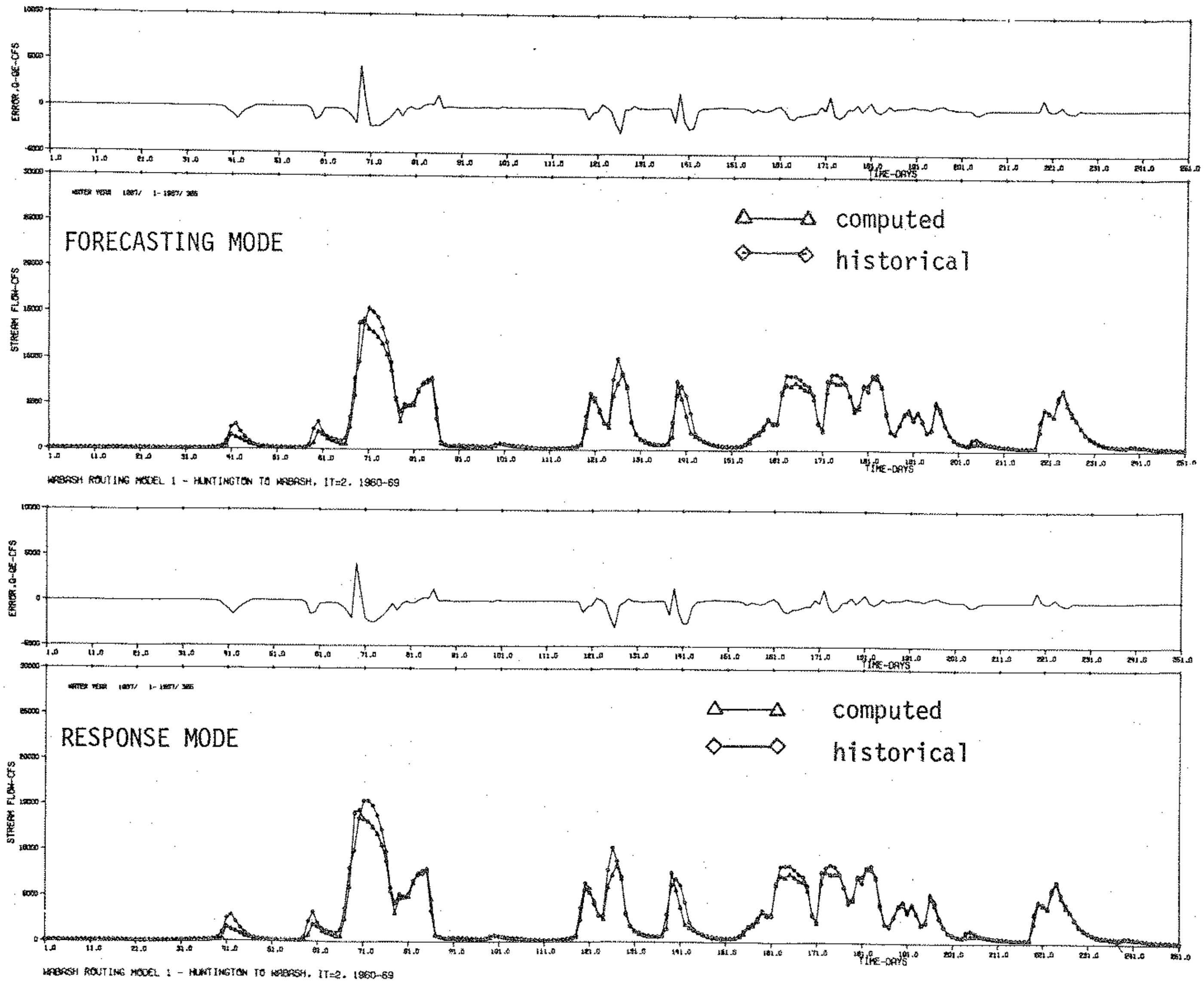


FIGURE V-7 *HISTORICAL REACH OUTFLOW AND THE RESULTS FROM MUSKINGUM MODEL OF REACH MODEL #1 FOR 1967 WATER YEAR*

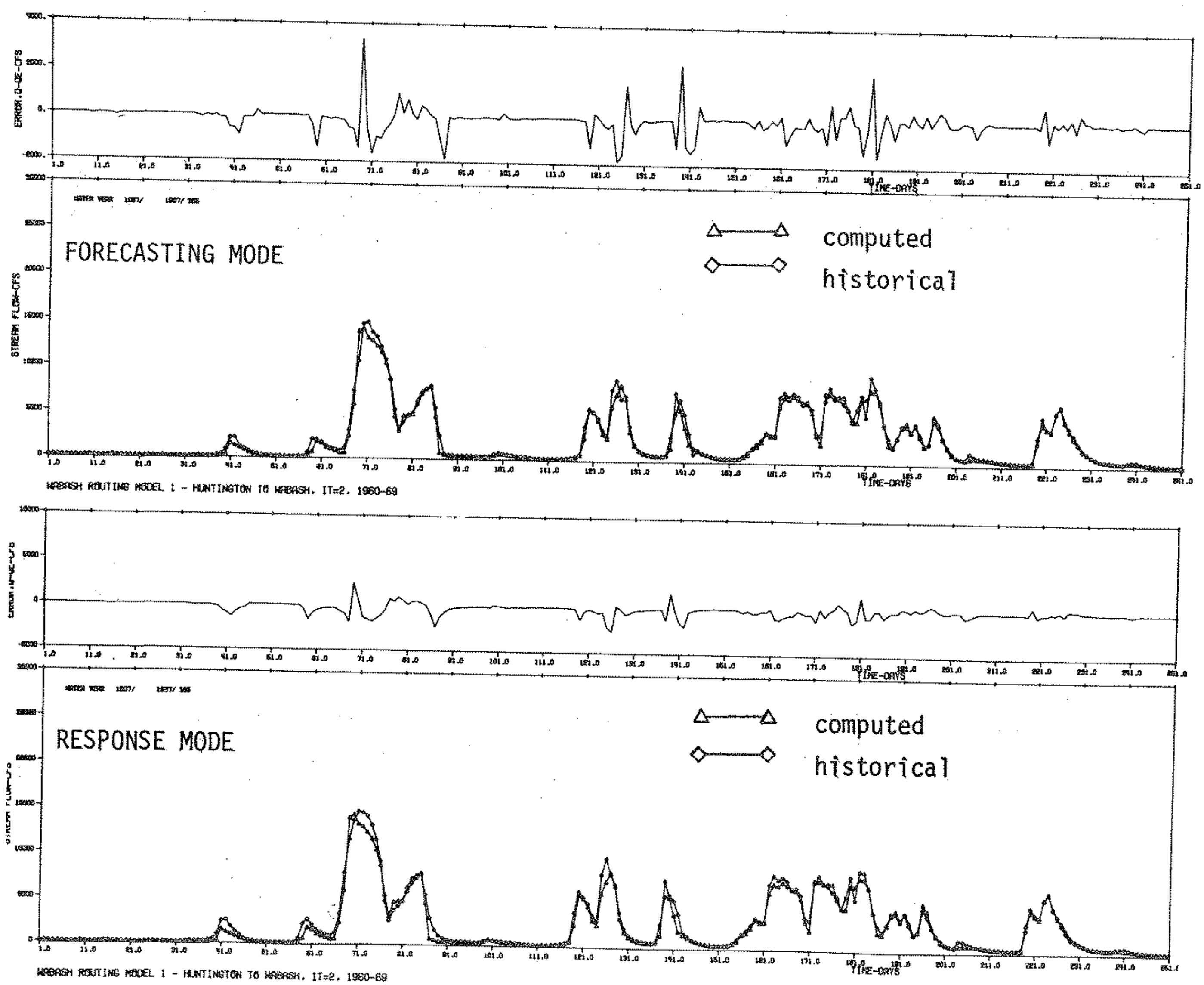


FIGURE V-8 HISTORICAL REACH OUTFLOW AND THE RESULTS FROM THE MILRM MODEL OF REACH MODEL #1 FOR 1967 WATER YEAR

The Tables V-2 and V-3 show (for the 1967 water year) the statistics of the historical outflow and the errors for both the Muskingum and MILRM models. The absolute errors (last column of Table V-2,3) are defined as the ratio of 100 MSE (i.e. mean square error) over MSS (i.e. mean square signal), where signal stands for outflow. Table V-4 shows the statistics of the historical outflow and the errors resulting from the use of models in both modes of use, for three reach models namely #1, #4 and #5, during the 10-year period of 1960 to 1969.

By comparing the error statistics shown in the Tables V-2 thru 4, the MILRM component performances are slightly superior to those of the Muskingum model in both modes of use. In particular the MILRM model predicts the peak flows better than the Muskingum model. This can be seen by inspection of the Figure V-7 and V-8.

10. DISCUSSION AND CONCLUSION

The MILRM and Muskingum models perform almost equally well as components of the Upper Wabash Simulation Model. The major reason for this similarity in performance is thought to be the fact that ungaged side inflow is relatively insignificant for all reach models. This fact can be inferred from the ratio of ungaged side inflow contributing areas to the contributing area of the outflow gaging station that bounds each of the six models. This ratio is fairly small for all models, ranging from 12% for model #1 to 7% for model #6. Thus each reach model approaches the situation for which Muskingum models were develped, namely for reaches without side inflow. Consequently, the merit of adding the rainfall data into the MILRM model seems to be limited in the present situation from the model component point of view. Nevertheless, it should be emphasized, that error statistics (rather than the plots in Figure V-7,8) indicate that the MILRM does perform consistently better than the Muskingum model.

Both the Muskingum and MILRM models are linear models. MILRM is a more generalized and has definite advantages over the Muskingum model.

First of all, though the number of parameters and variables in MILRM model components is greater, it is simpler than the Muskingum model when used as a building block in larger simulation models. No additional adjustments (using watershed area ratios) of computed or observed gaging

Model No.	Source	Model	Mean	Variance	Skewness	Kurtosis	Mean Square MSE, MSS	Absolute Error % $\frac{MSE}{MSS}$
Model 1	O(t)	historical	1687	7.06×10^{6}	2.24	8.08	9.90×10^{6}	
	Error	Muskingum	-107	2.79×10^{5}	-0.53	20.83	2.90×10^{5}	2.90
	Error	MILRM	-120	2.14×10^{5}	-1.121	12.07	2.29×10^{5}	2.30
Model 2	O(t)	historical	2606	1.43×10^{7}	1.91	5.99	2.11×10^{7}	
	Error	Muskingum	-38	2.20×10^{5}	-3.06	20.31	2.21×10^{5}	1.05
	Error	MILRM	34	1.84×10^{5}	-0.77	17.42	1.85×10^{5}	0.88
Model 3	O(t)	historical	3759	3.11×10^{7}	2.57	11.65	4.52×10^{7}	
	Error	Muskingum	-1.86	1.69×10^{5}	1.51	11.10	1.7×10^{5}	0.37
	Error	MILRM	-98	1.09×10^{5}	0.67	16.70	1.19×10^{5}	0.26
Model 4	O(t)	historical	3800	3.03×10^{7}	2.76	13.57	4.48×10^{7}	
	Error	Muskingum	-226	3.46×10^{5}	2.64	12.22	3.32×10^{5}	0.74
	Error	MILRM	12.7	1.95×10^{5}	1.72	37.4	1.95×10^{5}	0.43
Model 5	O(t)	historical	7212	8.34×10^{7}	2.61	12.2	1.35×10^{8}	
	Error	Muskingum	196	8.16×10^{5}	-1.78	24.3	8.54×10^{5}	0.63
	Error	MILRM	123	4.74×10^{5}	1.59	11.9	4.89×10^{5}	0.36
Model 6	O(t)	historical	7866	8.44×10^{7}	2.01	7.86	1.46×10^{8}	
	Error	Muskingum	-244	2.41×10^{6}	-6.24	57.2	2.47×10^{6}	1.69
	Error	MILRM	-315	1.24×10^{6}	-2.41	16.7	1.34×10^{6}	0.91

TABLE V-2 *COMPARISONS OF ERROR BETWEEN 2 MODELS: RESPONSE MODE 1967 WATER YEAR*

Model No.	Source	Model	Mean	Variance	Skewness	Kurtosis	Mean Square MSE, MSS	Absolute Error % $\frac{MSE}{MSS}$
1	O(t)	Historical	1687	7.06×10^6	2.24	8.08	9.90×10^6	
	Error	Muskingum	-93	2.53×10^5	0.22	25.8	2.60×10^5	2.64
	Error	MILRM	-62	1.87×10^5	0.96	20.37	1.91×10^5	1.93
2	O(t)	Historical	2606	1.43×10^7	1.91	5.99	2.11×10^7	
	Error	Muskingum	-36	2.06×10^5	-2.96	19.9	2.07×10^5	0.98
	Error	MILRM	15	1.37×10^5	0.72	17.8	1.38×10^5	0.65
3	O(t)	Historical	3759	3.11×10^7	2.57	11.65	4.52×10^7	
	Error	Muskingum	-2	1.63×10^5	1.50	11.2	1.63×10^5	0.36
	Error	MILRM	-44	8.05×10^4	0.51	17.7	8.24×10^4	0.18
4	O(t)	Historical	3800	3.03×10^7	2.76	13.57	4.48×10^7	
	Error	Muskingum	-226	2.81×10^5	-2.39	10.31	3.32×10^5	0.74
	Error	MILRM	4.3	1.43×10^5	5.11	81.9	1.43×10^5	0.32
5	O(t)	Historical	7212	8.34×10^7	2.61	12.2	1.35×10^8	
	Error	Muskingum	178	7.17×10^5	-1.78	24.5	7.49×10^5	0.55
	Error	MILRM	40	2.57×10^5	1.98	14.5	2.59×10^5	0.19
6	O(t)	Historical	7866	8.44×10^7	2.01	7.86	1.46×10^8	
	Error	Muskingum	-156	1.31×10^6	-6.79	69.8	1.33×10^6	0.91
	Error	MILRM	-171	6.54×10^5	-2.59	21.5	6.83×10^5	0.47

TABLE V-3 - COMPARISON OF ERRORS: FORECASTING MODE 1967 WATER YEAR

Mode	Model No.	Source	Model	Mean	Variance	Skewness	Kurtosis	Mean Square MSE, MSS	Error $\frac{MSE}{MSS}$ %
Response	1	O(t)	Historical	1212	5.51×10^6	3.48	17.4	6.98×10^6	
		Error	Muskingum	24	2.55×10^5	6.43	98.9	2.55×10^5	3.65
		Error	MILRM	-18	2.27×10^5	4.75	67.3	2.27×10^5	3.25
	4	O(t)	Historical	2939	2.32×10^7	3.76	21.6	3.18×10^7	
		Error	Muskingum	-102	7.70×10^5	4.38	120.5	7.81×10^5	2.45
		Error	MILRM	37	6.40×10^5	8.36	19.5	6.41×10^5	2.01
	5	O(t)	Historical	5633	5.98×10^7	3.50	18.9	9.16×10^7	
		Error	Muskingum	118	9.97×10^5	-1.95	81.4	1.01×10^6	1.1
		Error	MILRM	-49	8.99×10^5	0.08	82.1	9.02×10^5	0.98
Forecasting (one day ahead)	1	O(t)	Historical	1212	5.51×10^6	3.48	17.4	6.98×10^6	
		Error	Muskingum	21	2.17×10^5	6.24	98.2	2.17×10^5	3.11
		Error	MILRM	-9	1.43×10^5	3.13	53.2	1.43×10^5	2.05
	4	O(t)	Historical	2939	2.32×10^7	3.76	21.6	3.18×10^7	
		Error	Muskingum	-102	7.70×10^5	4.38	120.5	7.81×10^5	2.45
		Error	MILRM	13	3.62×10^5	8.85	244.9	3.62×10^5	1.14
	5	O(t)	Historical	5633	5.98×10^7	3.50	18.9	9.16×10^7	
		Error	Muskingum	108	8.81×10^5	-1.85	79.7	8.93×10^5	0.97
		Error	MILRM	-16	4.54×10^5	0.10	56.4	4.54×10^5	0.50

TABLE V-4 - ERROR COMPARISON FOR THE PERIOD 1960-1969 station data are needed when using MILRM components. The Muskingum model, on the other hand, requires some assumption on the basis of which to correct for ungaged side inflow, such as dividing flows on the basis of drainage area proportionality. Similar adjustments are needed to convert gaging station flows to flow at river junctions.

Secondly, if the model is to be used as a tool in the operation of surface water system, there will be the need for frequent updating of model parameters. The procedure to obtain the Muskingum parameters as outlined in PWRR #27 is quite laborious.

Thirdly, MILRM models have the advantage over a Muskingum model that it accepts rainfall data as an input and that it does so directly rather than via some unit hydrograph procedure. This permits a use of the model in the forecasting mode directly without further conversions. In the Muskingum model, forecasted rainfall has to be converted into forecasted discharge at the outlet of the reference watershed by some watershed model (e.g. unit hydrograph method).

The MILRM can then be recommended as the basic building block in composing large simulation models for use in planning studies and in real time operation of surface water systems.

VI. DATA BASE MANAGEMENT

1. *PURPOSE OF THE DATA BASE*

The database system built for the current deactivation effort is designed to allow users' programs to access climatological data and stream flow data in a fast, easy and relatively efficient manner. Advantages of the binary random format used presently include immediate access to needed word(s), and a degree of efficiency unobtainable by standard FORTRAN I/O routines. Also, if needed, a number of users may simultaneously access the data.

2. *STRUCTURE OF DATABASE*

The database (See Figure VI-1) consists of three basic parts. The first is the directory. The second is the main body of data. Both of these form the first file. Third, there is provision for overflow of data. The overflow is onto a second file. The user need not be concerned with the mechanics of overflow handling; the routines locate data from the files automatically.

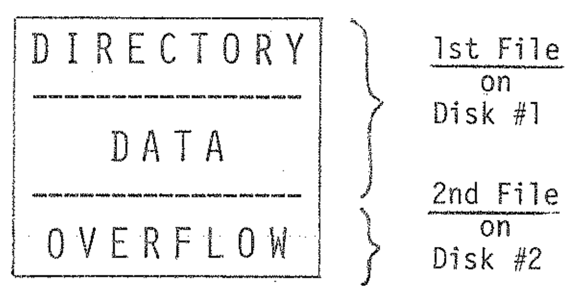


FIGURE VI-1 - DATA BASE STORAGE ALLOCATION

A. PREVIOUS STRUCTURE

The previous data base structure had three major deficiencies. These related to: (a) lack of error messages; (b) lack of indexing; and (c) limited directory size.

(a) The first difficulty was that error messages were non-existent. If the user made an error, the job would simply abort leaving a register dump and core dump, both of which are useful only to COMPASS programmers. Therefore, user-oriented error diagnostics were added to the read and index routines which inform the user of the precise nature of the error.

(b) A second difficulty was that the directory did not contain provision for indicating the last year of data in a given segment. As a result, indexing of the data base was difficult, and the number of years in a segment could only be obtained by a manual scan of a dump or a catalog of the data base file. This may seem minor, but a little thought will reveal that one must know what is on the data base in order to use it, and it is difficult to be sure of the contents of the data base without the aid of index routine(s). A random data base in a single record format is nearly useless without index routines giving the precise contents of the file. Many hours in routine clerical work were spent in determing the exact contents.

(c) Third, the directory had only five sectors which limited the amount of new data which could be added. This problem was alleviated by allowing for additional sectors to be added to the directory.

B. PRESENT STRUCTURE

In this section, the structure of the data base will be described in detail and the means used to correct the above noted deficiencies will be discussed.

A random access disk such as used for both the old and new data base work, is divided into tracks. Each track is divided into sectors. Each sector is 64 words long. System routines access a sector by its address. The address of the first sector is zero, the address of the second is one, and so on.

In the present data base, the first sectors of the first file (See Figure VI-1) contain the directory, while the remaining sectors contain the data. The actual number of directory sectors may possibly change from time to time, but these changes need not concern the user. Suppose the directory has n sectors. Then word 64n contains a pointer to the first empty data sector. Thus when data are added, the pointer is incremented to accommodate the new arrival. The number of Data Base Directory Entries (DBDE) which the directory will hold is the integer m, where: $m = (64n-1)/3$. (Note that any remainder which m may have is dropped). Thus the first 3m words of the directory consist of m 3-word DBDEs.

These 3-word DEDE's contain the following data:

1. The Data Base Reference (DBR) number. This is the number that the data base routines use to locate and identify segments.

2. First year of data in the segment.

3. Last year of data in the segment.

4. Address of the starting sector of the data for the segment.

5. Drainage area. This is greater than zero for flow data, -1 for temperature data, plus zero for precipitation data, -2 for evaporation data, and a minus zero (obtainable by leaving an integer data field blank) will allow for access of alternate segments. Alternate segments are seldom used, and are discussed only in the programmers' documentation.

6. "K" and "N" values used by other routines not discussed in this document.

7. USGS or NWS number of the station from which the data were obtained. If the value is negative, the value is NWS as opposed to USGS.

The overflow file consists entirely of data, and transition from one to the other need not concern the user. The remainder of this chapter will treat both files (See Figure VI-1) as one, because the data allocation between files is handled by the data base routines, and the user has no control over the matter.

Each sector of the data (64 words) is of a format which identifies precisely what the data is. (See Figure VI-2) The first two words are used for identification and the last 62 words contain data. The contents of these first two words will enable "lost" data to be identified and recovered. The contents of these two words follow:

1. Data Base Reference (DBR) number.
2. USGS (or NWS) number.
3. Water year of the data.
4. First day of data in the sector. (It takes 6 sectors of 62 data words to hold a year's data).
5. The number of days in the water year (365 or 366).

3. DATA BASE ACCESS

When a user requests access to data, the following sequence of events occurs. First, the user calls subroutine RANDOM which reads the directory

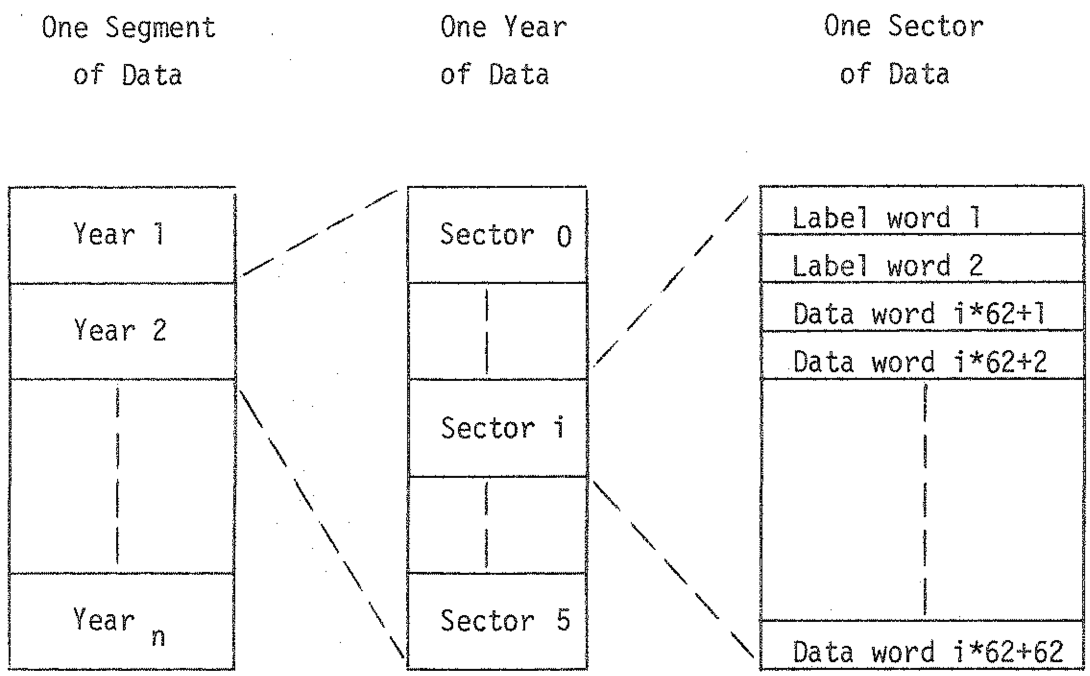


FIGURE VI-2 - SECTOR STRUCTURE

Note: the 6th sector will contain only 55 (or 56) words of data.

into memory, and sets up pointers. Then when the user requests a specific year of a specific segment n, the formula: 3(n-1) gives the address of the DBDE (See Figure VI-3) for that segment. The year is checked to see if it is between the first and last years given in the directory. Then the sector address of the data is fetched from the first word in the DBDE. Having this sector address, a call to the system random I/O routine will fetch the data sector. The year and segment as given in the first two words of the data are checked against those requested. If a mis-match occurs, an error message will be issued to the DAYFILE and the job step aborts. Otherwise the data will be loaded to the user's buffer and the routine will return to the caller.

In the event of missing data, the value 999.00 for temperature, 9.99 for evaporation, and 99.99 for precipitation and 0 for discharge data will be inserted in place of the missing data. Missing discharge data will also have a -1 in the ND (Number days in the water year) parameter.

This access method is reasonably fast, uses little extra storage, and provedes for easy user access to the data. To obtain a listing of the contents of the data base, one need only run an indexing job as described in Chapter VII, section 2A. Since the last water year of data in the segment is given, and the data base is of a uniform format, indexing is easy and will yield complete information.

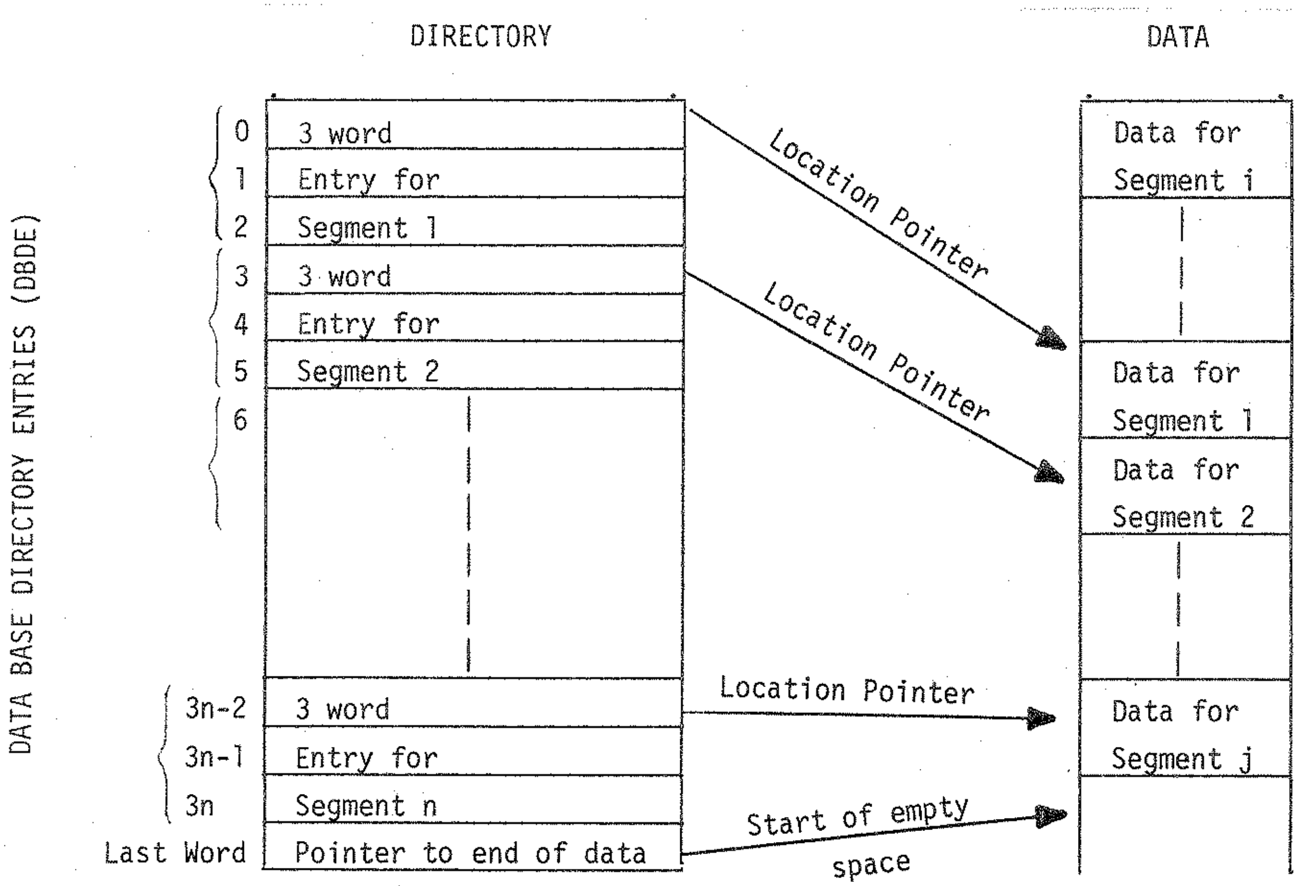


FIGURE VI-3 - DIRECTORY - DATA RELATIONSHIP

NOTE: actual location of Data Blocks is arbitrary.

VII. DATA BASE USERS' MANUAL

1. *GENERAL INFORMATION*

This section will contain information on reading and indexing the data base only. Writing of data onto the data base is best left to a competent COMPASS (CDC assembly language) programmer; information on the writing routines is found in the Wabash Data Base Programmer's Manual.

All of the data base routines are stored in a library in PFILES. The routines of interest on the library are:

RNDSYS
WABINDX
WABCAT

The library may be accessed by the following control card:

```
PFILES(,WABLIB,ID=BXL)
```

Loading of WABINDX and WABCAT is discussed in Sections VII-2-A,B as they are "stand alone" programs, and use a different access procedure than the RNDSYS routines.

RNDSYS is a large routine which contains many entry points, each of which acts just like a small subprogram. The combination of each of the simpler subprograms into the large RNDSYS routine saves space, and is more efficient.

To access any of the RNDSYS routines, one uses the following control card sequence:

```
PFILES(,WABLIB,ID=BXL)
LOADX(LGO,WABLIB,USEROBJ,RUNLIB2,RUNLIB,MNFLIB)
```

It is assumed that the user has already accessed the data base (see below), and compiled his own programs, writing the relocatable binary to file LGO. It is vital that USEROBJ be included in the library list. However, the user may not require all of the systems libraries RUNLIB2, RUNLIB, and MNFLIB. To access the data base itself, use the following control card:

```
XEQ(WABGET,ID=BXL)
```

Details on use of the control cards will follow in Section VII-2 and 3.

2. *UTILITY ROUTINES*

A. DIRECTORY INDEX

Routine WABINDX produces a short index listing of the data base directory. A sample run of WABINDX is found in Appendix A. The variables shown on the listing in Appendix A are:

NUMBER	Data Base Reference Number. This is the internal identification number used by all of the data base routines to identify segments.
USGS NO	USGS or NWS number of the station from which the data were obtained. If this number is positive, then the number represents the USGS number of the station. Of course, this also means that the source of the data was the United States Geological Survey. If USGS NO is negative, then the absolute value of the number represents the NWS number of the station. Of course, this means that the data were obtained from the National Weather Service.
SECT ADR	Address of the first sector of data. This information is mostly of value to the systems programmer.
N*1000	Used by routine GENQ (gamma function unit hydrograph; see earlier Wabash Simulation Model Reports).
K*1000	Also used by GENQ.
WS AREA	Watershed area. This variable serves two functions. First, if the segment in question contains flow data, this number represents the area in square miles of the watershed in which the station is located. Second, if the data is not flow data, then this variable indicates the type of data in the segment. The type will be zero for precipitation segments; -1 for temperature segments; -2 for evaporation segments; and -0 for alternate segments. Alternate segments are segments that are not in the data base but are accessed through RNDSYS. Alternate segments are seldom used and are discussed only in the Wabash Data base Programmer's Manual.
FIRST YR	First water year of data.
LAST YR	Last water year of data.
ALT STN	Number of the alternate segment

Finally, the number of data sectors in the data base is given at the end of the index.

To run the index routine, use the following control card sequence:

```
JOB with CM15000,T4,TU1000.
PASS=
PFILES(,WABLIB,ID=BXL)
XEQ(WABGET,ID=BXL)
LINK(F=WABLIB,E=WABINDX,P=USEROBJ,X)
6/7/8/9 (End of Information)
```

A sample run of WABINDX against the current Wabash data base is found in Appendix A.

B. CATALOG OF DATA BASE

The routine WABCAT produces a very detailed catalog of the data base data. A sample run of WABCAT is found in Appendix B.

The variables shown on the listing in Appendix B are:

DSECT	Number of the data block (i.e. a sector). In order to find the address of the sector in which the block resides, simply add k to the block number. (Where k is the number of directory sectors).
DBR	Data Base Reference Number.
TYPE	Type of data in the block.
WYR	Water year of the data.
NOD	Number of days in water year WYR
NUSGS	USGS (or NWS) number of the station from which the data were obtained.
SDY	Starting day in the block. This is the day in the water year which is first in the block of 62 words. Note that (barring errors) SDY will always be one of the following: 1, 63, 125, 187, 249 or 311.

Following this, is an octal representation of the label words, and a listing of the first five data words. The label words will be of interest mainly to the systems programmer, but the listing of the data words will prove useful for checking data quality, etc.

To use WABCAT, use the following control card sequence:

```
JOB with CM15000,T60,TU10000,L10048.
PASS=
PFILES(,WABLIB,ID=BXL)
XEQ(WABGET,ID=BXL)
LIBRARY(RUNLIB2)
LINK(F=WABLIB,E=GRBCAT,P=USEROBJ,X)
6/7/8/9 (End of Information)
```

Note that if E=WABCAT2 is specified on the LINK card instead of E=WABCAT, then the overflow file will be catalogued instead of the main file. Partial listings of jobs using WABCAT and WABCAT2 are to be found in Appendix B.

C. RNDSYS UTILITY ROUTINES

There are several routines which return a single value, such as the USGS number, or set various parameters. A list of these routines and their uses follows.

RANDOM

RANDOM must be called before any other data base routine is used. If this procedure is not followed, the routine(s) simply will not work. To use any data base routine, simply include the statement:

```
CALL RANDOM
```

in the main program. The routines WABINDX and WABCAT are exceptions to this rule, since they are complete within themselves.

GETID

This function returns the USGS (or NWS) number of a segment. Note that this function must be declared type INTEGER in a FORTRAN program. To use function GETID, simply call it with one parameter: (NOS). NOS is the Data Base Reference Number. For example, to set variable I to the USGS number of flow segment 16, use the FORTRAN statement:

```
I=GETID(16)
```

Note that one must have:

```
INTEGER GETID
```

or the equivalent in the program declarations.

GETN and GETK

The purpose of these two functions is to obtain two parameters of a segment, N and K, which are used by GENQ. GETK obtains the K-value, and GETN obtains the N-value. For example, to obtain the K-value of segment 16, one might use:

```
VALUEK=GETK(16)
```

Only one parameter (Data Base Reference Number) is used.

GETDA

Function GETDA is a function which returns the watershed area (drainage area) pertaining to the segment given in the single parameter. For example:

```
WAREA=GETDA(16)
```

will put the watershed area of segment 16 into variable WAREA.

SETYR

Subroutine SETYR is used to set the year being worked with to a given value. One of its purposes is in conjunction with some of the read routines. In the section dealing with read routines, this will be discussed. SETYR is also used in conjunction with GENQ (a routine not discussed in this document) and an example follows.

```
CALL SETYR(1972,RFV,HPU)
```

This call will set the current year to 1972, and will set the RFV and HPU values. These two values are used by GENQ and their purpose will not be discussed here. However, these values are optional, if GENQ is not used; the form: (1972) is quite valid for use with read routines.

SETN and SETK

SETN and SETK are subroutines which set the N and K values of segment NOS. Examples of use are:

```
CALL SETN(16,2536)
CALL SETK(16,1476)
```

The above sequence will set N to 1.536, and K to 1.476. Note that the values given are integer values and are 1000 times the actual value.

3. *READING DATA FROM THE DATA BASE*

Data may presently be read from the data base in two different formats: One day's data at a time, or an entire year's data at a time.

In order to use any of the RNDSYS routines, there are two essential requirements that a FORTRAN job must meet in order to use the data base. The first is that the data base must be accessible to the executing job. This requirement is met by the XEQ(WABGET,ID=BXL) control card mentioned earlier.

The other requirement is that subroutine RANDOM be called PRIOR to calling ANY other data base routines. If this is not done, it is impossible to access either directory or data.

When any of the read routines return a missing value, the missing value will show up as: 99.99 for missing precipitation, 999.0 for missing temperature, and 9.99 for missing evaporation. Missing flow data is indicated by zero, and the number of days value (ND) is returned as -1.

An example program with output will be found in Appendix C. Figure VII-I shows the general procedure for reading from the Data Base.

A. DATA RETRIEVAL FUNCTIONS: STREAM, RAIN, TEMPER AND EVAP

Data Retrieval Functions read one word of data per call. The function is called by giving it the Data Base Reference Number and the day in the water year requested. In order to specify the water year required, the user must call subroutine SETYR, which (for the purpose of Data Retrieval Functions) takes one parameter, the year requested. The given year stays current until SETYR is called again with a different value. For example, to set the current water year to 1973, one simply uses:

```
CALL SETYR(1973)
```

The following table shows the name of the Data Retrieval Functions and the type of data that they retrieve. In the table NOS is the Data Base Reference Number and NOD is the day in the water year. Note that all functions are type REAL functions.

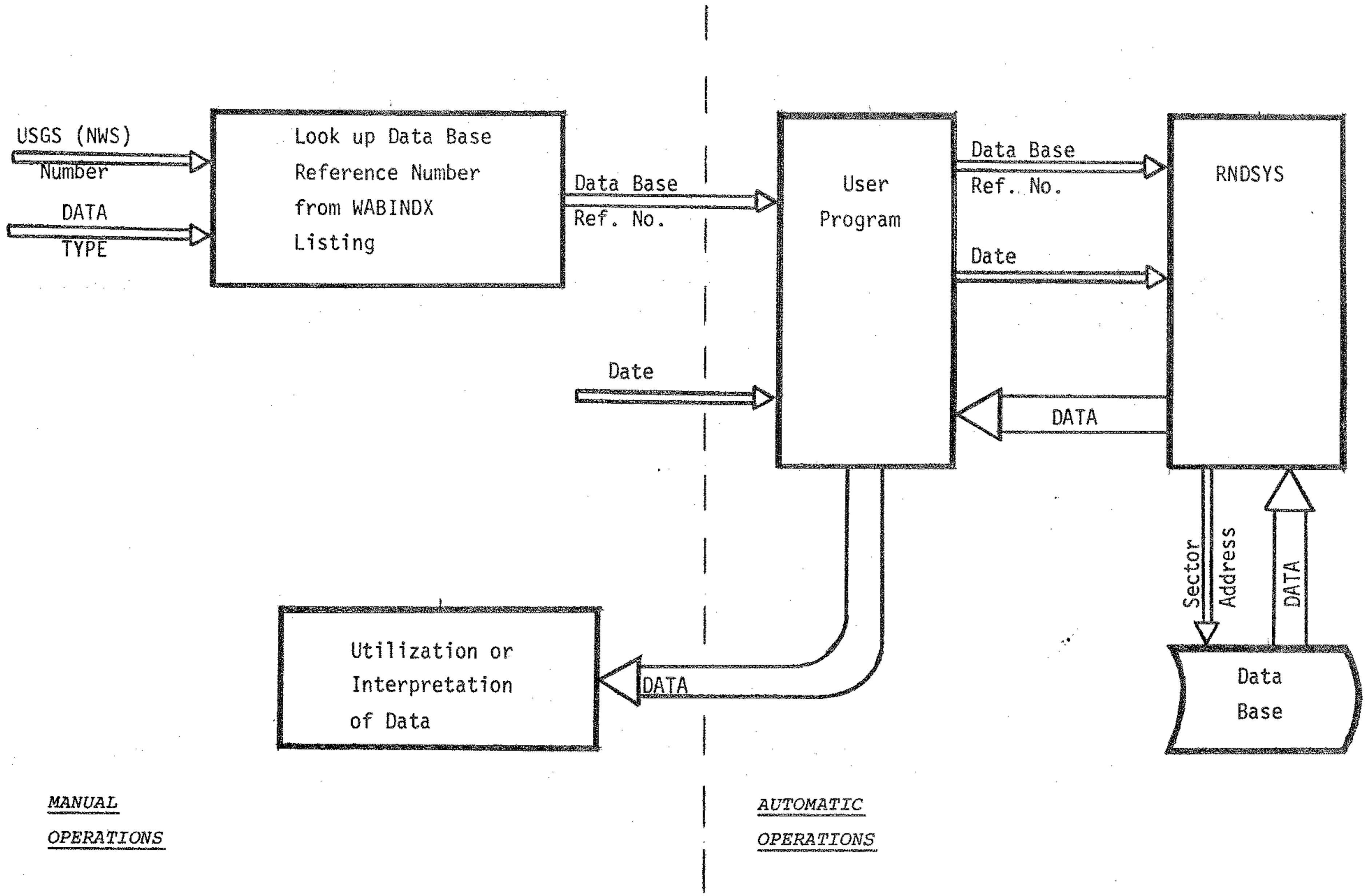


FIGURE VII-1 - PROCEDURE FOR READING DATA FROM DATA BASE

STREAM(NOS,NOD) Returns stream flow data
RAIN(NOS,NOD) Returns precipitation data
TEMPER(NOS,NOD) Returns temperature data
EVAP(NOS,NOD) Returns evaporation data

For example, if a user wishes to find the temperature for segment 148 on Oct. 4, 1972, then the following FORTRAN statements will suffice:

```
CALL SETYR(1972)
TEMP=TEMPER(148,4)
```

B. DATA RETRIEVAL IN YEARLY BLOCKS: RDDATA

Subroutine RDDATA is a data base routine designed to retrieve and return blocks of data in the form of an array of 366 words. It is the responsibility of the user to be certain that word 366 is not used except in leap years. To call RDDATA, use the FORTRAN statement:

```
CALL RDDATA(NOS,IYR,B,ND,IT)
```

where:

NOS is the Data Base Reference Number.
IYR is the water year (The year set by SETYR has no effect).
B is a real array of length 366, and is returned by RDDATA.
ND is the number of days in water year IYR (returned by RDDATA).
IT is the type code. The type code is: 1 for flow data, 0 for precipitation data, -1 for temperature data, and -2 for evaporation data.

For example, let a user wish to retrieve the precipitation data for 1968, segment 120, and have the data go to real array RAINY. Then the correct FORTRAN call is:

```
CALL RDDATA(120,1968,RAINY,ND,0)
```

Caution: a call to RDDATA will replace the year set by subroutine SETYR with the value given RDDATA. Therefore, when intermixing calls to RDDATA and any of the Data Retrieval Functions, the user should exercise caution.

4. *RNDSYS ERROR MESSAGES*

There are six error messages which may be generated by RNDSYS. Each message is followed by traceback information, which will enable the user to determine where in his program the error occurred. These messages are of the same format as messages generated by Purdue MACE system FORTRAN callable routines. Information given includes the nature of the error, its number, the routine that detected it, and traceback information. The traceback information lists the routine which called the error detecting routine, then the routine which called it, and so on until the main program is reached. For each routine, the absolute and relative addresses of the calling locations are given. The relative location given will enable the user to locate, from the compiler source listing, the statement which called the error detecting data base routine.

Following is a list of error messages, and their meanings and possible causes:

ERROR 111 -- DIRECTORY NOT FOUND

This error is detected by RANDOM and is caused by not having the data base accessible to the user's job.

ERROR 112 -- SEGMENT xxxx LESS THAN yyyy OR GREATER THAN zzzz

An attempt was made to access a segment whose number was less than the minimum or greater than the maximum set for the type of segment being referenced.

ERROR 113 -- SEGMENT xxxx NOT FOUND

Caused by trying to access a non-existent segment. This error is detected after the test for range validity (error 112). Thus it is generated when the number itself is legal, but no segment of that number actually exists. Another cause is neglecting to call RANDOM before calling the routine which detected this error.

ERROR 114 -- YEAR xxxx NOT FOUND

An attempt was made to access a non-existent year. This error is often made by specifying "72" for example, instead of "1972".

ERROR 115 -- DAY xxxx NOT BETWEEN 1 AND 366

Caused by illegal day when calling STREAM, RAIN, TEMPER, or EVAP. No check is made to determine that day 366 is not used in non-leap years.

ERROR 116 -- DATA RETRIEVAL ERROR

Just before the requested data is returned to the user on read operations, a final check is made to ascertain that the year and segment requested match the information on the label words, and if the check fails, error 116 is generated. This indicates a serious data base, RNDSYS, or operating system error, and should be reported to the systems programmer in charge of the data base.

VIII. SUMMARY AND CONCLUSIONS

During the 1969-74 period a series of projects sponsored by the Purdue Water Resources Center, resulted in a large computer simulation model for the Upper Wabash River and Reservoir System. The work reported herein was designed:

(a) to deactivate the simulation system in such a manner that the model can be resurrected with a minimal amount of effort;

(b) to reorganize the project data base, and provide efficient access routines;

(c) to bring into the data base all stream flow data and precipitation data that are needed to extend the simulation model down to the Ohio River;

(d) to replace the Muskingum Routing components of the Wabash Simulation model by a more accurate and general model component;

(e) to find a solution for the fact that it would be difficult from the point of view of computation time to extend the current simulation model down to the Ohio River.

Each of these aims was accomplished. The results and conclusions may be summarized as follows:

1. A data base was built in a binary, random format consisting of a directory, a main body of data, and an overflow file that permits data base manipulations. This organization and the binary, random format allow (given appropriate access routines) immediate access to needed words and the utilization of the efficiency obtainable by standard FORTRAN I/O routines.
2. A Data Base Users Manual was prepared that facilitates and makes for efficient reading, indexing, cataloging of daily, monthly, and yearly year data samples. In addition some error diagnostics were built in.
3. The available daily precipitation data for the entire state of Indiana were brought into the data base. Also the stream flow data needed to extend the Upper Wabash Simulation Model in the downstream direction were added into the Data Base.

4. The Muskingum routing components were replaced by Multi-input Linear Routing Models (MILRM's). Taken as a single component the advantage of an MILRM in predicting today's reach outflow is not very obvious. Each of the six built MILRM's performs consistently better than individual Muskingum components. This is true particularly for the peak flows. However, the differences must be determined statistically. They seem marginal when one inspects the graphical comparisons of component model outputs.
5. A major merit of the new MILRM components is that it solves the problem of extending or enlarging the Wabash Simulation model. The MILRM components are considerably more efficient. They also accept precipitation data directly thus eliminating the need for hydrologic sub-models for ungaged contributing watershed. Finally, the problem of operating the simulation model in a forecasting mode that uses precipitation forecasts has been solved.

IX. APPENDICES

```
22.35.29.      ,S24,CM15000,T4,TU500.
22.35.29.XEQ(WABGET,ID=BXL)
22.35.30. WABGET          12 WORDS
22.35.30. PFILES COMPLETE.
22.35.31.DISABLE(PLIST)
22.35.33.PFILES(,WABLIB,ID=BXL)
22.35.35. WABLIB      1723 WORDS
22.35.35. PFILES COMPLETE.
22.35.35.LINK(E=WABINDX,F=WABLIB,X,P=USEROBJ)
22.35.36.CX        .165 SEC., NL  15200 WORDS
22.35.37.CX        .237 SEC., NL   3300 WORDS
22.35.39.MAX TRACKS:  4
22.35.39.CP SECS:       0.338 = $ 0.01
22.35.39.IO UNITS:        211 = $ 0.03
22.35.39.LINES:           183 = $ 0.03
22.35.39.TOTAL COST ESTIMATE = $ 0.07
```

<u>APPENDIX A</u> - *CONTROL CARDS AND OUTPUT OF* WABINDX. *THE OUTPUT SHOWS THE CONTENTS OF THE WABASH DATA BASE IN A HYDROLOGIC DATA RECORD FASHION.*

******* DATABASE DIRECTORY ******* 07/19/76.

NUMBER	USGS NO	SECT ADR	N*1000	K*1000	WS AREA	FIRST YR	LAST YR	ALT STN
1	3322000	8	4900	1100	107000	1941	1973	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
4	3323000	206	0	0	532	1931	1971	2
5	3323500	452	5129	800	721	1952	1973	4
6	3324000	584	6460	559	263	1945	1973	0
7	3324300	758	0	0	425	1958	1973	0
8	3324500	854	6060	570	557	1925	1973	7
9	3325000	1148	0	0	1768	1924	1973	0
10	3326500	1448	0	0	682	1924	1973	0
11	3327000	1748	5539	679	808	1953	1973	10
12	3327500	1874	0	0	2682	1944	1973	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
15	3328500	2054	4650	820	789	1944	1973	14
16	3329000	2234	5300	1000	3777	1924	1973	0
17	3329500	2534	0	0	4072	1941	1971	0
18	3329700	2720	0	0	274	1955	1973	24
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
22	3333000	2834	4080	1369	1865	1940	1973	21
0	0	0	0	0	0	0	0	0
24	3334000	3038	0	0	396	1945	1973	23
25	3334500	3212	0	0	242	1945	1973	24
26	3335000	3386	5379	789	794	1955	1973	24
27	3335500	3500	0	0	7267	1924	1973	0
28	3335700	3800	5219	769	323	1956	1973	25
29	3336000	3908	5270	1109	8208	1940	1973	0
30	3339000	4112	0	0	1279	1929	1973	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
33	3339500	4382	0	0	509	1939	1973	0
34	3340000	4592	0	0	670	1941	1971	33
35	3340500	4778	0	0	11100	1928	1973	0
36	3340800	5054	0	0	132	1958	1973	0
37	3340900	5150	0	0	215	1957	1973	36
38	3341300	5252	0	0	440	1957	1973	37
39	3341500	5354	5000	1000	12200	1928	1973	0
40	3342000	5630	0	0	13100	1940	1973	0
0	0	0	0	0	0	0	0	0
42	3342500	5834	0	0	228	1944	1973	41
43	3343000	6014	0	0	13700	1931	1973	0
44	3345500	6272	0	0	1513	1915	1973	0
45	3346000	6626	0	0	319	1941	1973	0
46	3361000	6824	0	0	184	1951	1973	0
47	3361500	6962	0	0	421	1944	1973	46
48	3362000	7142	0	0	107	1944	1973	0
49	3362500	7322	0	0	474	1944	1973	0
50	3363000	7502	0	0	1060	1942	1973	0
51	3363500	7694	0	0	303	1931	1973	0
52	3363900	7952	0	0	534	1968	1973	51
53	3364000	7988	0	0	1707	1949	1973	0
54	3364500	8138	0	0	91	1949	1973	0
55	3365000	8288	0	0	155	1949	1973	0
56	3365500	8438	0	0	2341	1928	1973	0

APPENDIX A - <u>WABINDX</u> *(cont'd)*

******* DATABASE DIRECTORY ******* 07/19/76.

NUMBER	USGS NO	SECT ADR	N*1000	K*1000	WS AREA	FIRST YR	LAST YR	ALT STN
57	3366500	8714	0	0	293	1949	1973	0
58	3369500	8864	0	0	198	1940	1973	0
59	3371500	9068	0	0	3861	1958	1973	0
60	3372500	9164	0	0	432	1956	1973	0
61	3373000	9272	0	0	573	1940	1971	60
62	3373500	9464	0	0	4927	1924	1973	0
63	3373700	9764	0	0	287	1966	1973	88
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
71	3351000	9812	0	0	1219	1931	1973	0
72	3351500	10070	0	0	169	1942	1973	0
73	3352500	10262	0	0	298	1930	1973	72
74	3353000	10526	0	0	1635	1931	1973	0
75	3353200	10784	0	0	103	1958	1973	0
76	3353500	10880	0	0	174	1940	1973	75
77	3353800	11084	0	0	212	1958	1973	0
78	3354000	11180	0	0	2444	1948	1973	0
79	3356000	11336	0	0	100	1947	1973	0
80	3357000	11498	0	0	2988	1926	1971	0
81	3357500	11774	0	0	326	1950	1973	0
82	3358000	11918	0	0	245	1950	1973	0
83	3359000	12062	0	0	294	1940	1973	82
84	3360000	12266	0	0	830	1932	1973	0
85	3360500	12518	0	0	4688	1929	1973	0
86	3374000	12788	0	0	11125	1938	1973	0
87	3374500	13004	0	0	171	1962	1973	0
88	3375500	13076	0	0	262	1949	1973	87
0	0	0	0	0	0	0	0	0
90	3376500	13226	0	0	822	1941	1973	88
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
100	-120676	13424	0	0	0	1949	1973	0
101	-120824	13574	0	0	0	1949	1969	0
102	-120830	13700	0	0	0	1970	1973	0
103	-121147	13724	0	0	0	1952	1969	0
104	-121415	13832	0	0	0	1949	1951	0
105	-121739	13850	0	0	0	1949	1973	0
106	-121952	14000	0	0	0	1959	1973	0
107	-122149	14090	0	0	0	1949	1973	0
108	-122825	14240	0	0	0	1950	1973	0
109	-123037	14384	0	0	0	1949	1973	0
110	-123062	14534	0	0	0	1949	1971	0
111	-123082	14672	0	0	0	1949	1973	0
112	-123087	14822	0	0	0	1949	1951	0

APPENDIX A - WABINDX *(cont'd)*

******* DATABASE DIRECTORY ******* 07/19/76.

NUMBER	USGS NO	SECT ADR	N*1000	K*1000	WS AREA	FIRST YR	LAST YR	ALT STN
113	-123777	14840	0	0	0	1949	1973	0
114	-124176	14990	0	0	0	1949	1973	0
115	-124181	15140	0	0	0	1949	1951	0
116	-124667	15158	0	0	0	1949	1968	0
117	-124715	15278	0	0	0	1955	1973	0
118	-125117	15392	0	0	0	1949	1973	0
119	-125122	15542	0	0	0	1957	1973	0
120	-125337	15644	0	0	0	1949	1973	0
121	-125837	15794	0	0	0	1949	1973	0
122	-126864	15944	0	0	0	1949	1951	0
123	-127069	15962	0	0	0	1949	1973	0
124	-127482	16112	0	0	0	1949	1973	0
125	-127601	16262	0	0	0	1949	1951	0
126	-127747	16280	0	0	0	1949	1973	0
127	-128789	16430	0	0	0	1949	1969	0
128	-129138	16556	0	0	0	1949	1973	0
129	-129240	16706	0	0	0	1949	1972	0
130	-129424	16850	0	0	0	1949	1973	0
131	-129430	17000	0	0	0	1949	1973	0
132	-129670	17150	0	0	0	1949	1973	0
133	-129678	17300	0	0	0	1949	1973	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
138	-120676	17450	0	0	-1	1949	1973	0
139	-120824	17600	0	0	-1	1958	1969	0
140	-120830	17672	0	0	-1	1970	1973	0
141	-121739	17696	0	0	-1	1964	1973	0
142	-121952	17756	0	0	-1	1959	1973	0
143	-122149	17846	0	0	-1	1949	1973	0
144	-122825	17996	0	0	-1	1963	1973	0
145	-123037	18062	0	0	-1	1949	1973	0
146	-123062	18212	0	0	-1	1949	1971	0
147	-123082	18350	0	0	-1	1949	1973	0
148	-123777	18500	0	0	-1	1960	1973	0
149	-124176	18584	0	0	-1	1949	1973	0
150	-124667	18734	0	0	-1	1949	1968	0
151	-124715	18854	0	0	-1	1955	1973	0
152	-125122	18968	0	0	-1	1957	1973	0
153	-125337	19070	0	0	-1	1949	1973	0
154	-127482	19220	0	0	-1	1949	1973	0
155	-127747	19370	0	0	-1	1949	1973	0
156	-129138	19520	0	0	-1	1949	1973	0
157	-129240	19670	0	0	-1	1949	1972	0
158	-129424	19814	0	0	-1	1965	1973	0
159	-129430	19868	0	0	-1	1954	1973	0
160	-129670	19988	0	0	-1	1949	1973	0
161	-129678	20138	0	0	-1	1949	1973	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
164	-121952	20288	0	0	-2	1961	1973	0
165	-129430	20366	0	0	-2	1957	1973	0
0	0	0	0	0	0	0	0	0

TOTAL NUMBER OF SECTORS CONTAINING DATA = 20460 0 0 0 0
0 0 0 0

APPENDIX A – WABINDX *(cont'd)*

```
22.35.33.      ,S24,CM15000,T72,TU5000.
22.35.33.XEQ(WABGET,ID=BXL)
22.35.36. WABGET         12 WORDS
22.35.36. PFILES COMPLETE.
22.35.37.DISABLE(PLIST)
22.35.39.PFILES(,WABLIB,ID=BXL)
22.35.49. WABLIB       1723 WORDS
22.35.49. PFILES COMPLETE.
22.35.49.LIBRARY(RUNLIB2)
22.35.50.LINK(E=WABCAT,F=WABLIB,X,P=USEROBJ)
22.35.57.CX         .169 SEC., NL  15200 WORDS
22.36.00.CX         .347 SEC., NL  22200 WORDS
22.36.01.CX         .412 SEC., NL   6400 WORDS
22.36.18.MAX TRACKS:  7
22.36.19.CP SECS:         7.164 = $ 0.34
22.36.19.IO UNITS:          948 = $ 0.15
22.36.19.LINES:             960 = $ 0.19
22.36.19.TOTAL COST ESTIMATE = $ 0.68
```

<u>APPENDIX B</u> - *CONTROL CARDS AND SAMPLE OUTPUT OF* WABCAT. *THE OUTPUT SHOWS THE CONTENTS OF THE WABASH DATA BASE IN A SECTOR BY SECTOR FASHION.*

Note: the total output would take 600 pages; only 2 pages have been shown in App. B.

CATALOG OF BINARY DATABASE 07/19/76

DSECT	DBR	TYPE	WYR	NOD	NUSGS	SDY	LABEL WORDS		FIRST FIVE DATA WORDS				
1	1	FLOW	1941	365	3322000	1	00000001003625000555	14530220000000000001	19000.00	19000.00	19000.00	19000.00	19000.00
2	1	FLOW	1941	365	3322000	63	00000001003625000555	14530220000000000077	88000.00	97600.00	100000.00	90000.00	75400.00
3	1	FLOW	1941	365	3322000	125	00000001003625000555	14530220000000000175	186000.00	168000.00	149000.00	128000.00	110000.00
4	1	FLOW	1941	365	3322000	187	00000001003625000555	14530220000000000273	90000.00	101000.00	117000.00	130000.00	136000.00
5	1	FLOW	1941	365	3322000	249	00000001003625000555	14530220000000000371	106000.00	104000.00	121000.00	160000.00	200000.00
6	1	FLOW	1941	365	3322000	311	00000001003625000555	14530220000000000467	15000.00	15000.00	15000.00	15000.00	15000.00
7	1	FLOW	1942	365	3322000	1	00000001003626000555	14530220000000000001	15000.00	15000.00	15000.00	15000.00	15000.00
8	1	FLOW	1942	365	3322000	63	00000001003626000555	14530220000000000077	34000.00	34000.00	34000.00	34000.00	38000.00
9	1	FLOW	1942	365	3322000	125	00000001003626000555	14530220000000000175	74100.00	70400.00	69800.00	69000.00	111000.00
10	1	FLOW	1942	365	3322000	187	00000001003626000555	14530220000000000273	125000.00	119000.00	113000.00	116000.00	154000.00
11	1	FLOW	1942	365	3322000	249	00000001003626000555	14530220000000000371	58000.00	58000.00	58000.00	58000.00	58000.00
12	1	FLOW	1942	365	3322000	311	00000001003626000555	14530220000000000467	59000.00	59000.00	89600.00	98700.00	108000.00
13	1	FLOW	1943	365	3322000	1	00000001003627000555	14530220000000000001	50000.00	60000.00	69500.00	69800.00	55000.00
14	1	FLOW	1943	365	3322000	63	00000001003627000555	14530220000000000077	195000.00	195000.00	201000.00	213000.00	229000.00
15	1	FLOW	1943	365	3322000	125	00000001003627000555	14530220000000000175	191000.00	227000.00	252000.00	273000.00	286000.00
16	1	FLOW	1943	365	3322000	187	00000001003627000555	14530220000000000273	213000.00	191000.00	167000.00	145000.00	120000.00
17	1	FLOW	1943	365	3322000	249	00000001003627000555	14530220000000000371	189000.00	188000.00	190000.00	186000.00	180000.00
18	1	FLOW	1943	365	3322000	311	00000001003627000555	14530220000000000467	80000.00	88500.00	90000.00	106000.00	118000.00
19	1	FLOW	1944	365	3322000	1	00000001003630000555	14530220000000000001	9000.00	9000.00	9000.00	9000.00	9000.00
20	1	FLOW	1944	365	3322000	63	00000001003630000555	14530220000000000077	17000.00	17000.00	17000.00	17000.00	21000.00
21	1	FLOW	1944	365	3322000	125	00000001003630000555	14530220000000000175	77900.00	77900.00	80500.00	85000.00	37000.00
22	1	FLOW	1944	365	3322000	187	00000001003630000555	14530220000000000273	293000.00	258000.00	223000.00	195000.00	184000.00
23	1	FLOW	1944	365	3322000	249	00000001003630000555	14530220000000000371	80000.00	80000.00	80000.00	50900.00	31000.00
24	1	FLOW	1944	365	3322000	311	00000001003630000555	14530220000000000467	11000.00	11000.00	11000.00	11000.00	12000.00
25	1	FLOW	1945	365	3322000	1	00000001003631000555	14530220000000000001	34000.00	34000.00	34000.00	34000.00	34000.00
26	1	FLOW	1945	365	3322000	63	00000001003631000555	14530220000000000077	50000.00	50000.00	50000.00	50000.00	45000.00
27	1	FLOW	1945	365	3322000	125	00000001003631000555	14530220000000000175	50000.00	50000.00	50000.00	50000.00	60000.00
28	1	FLOW	1945	365	3322000	187	00000001003631000555	14530220000000000273	335000.00	303000.00	278000.00	267000.00	265000.00
29	1	FLOW	1945	365	3322000	249	00000001003631000555	14530220000000000371	78600.00	70000.00	84400.00	89600.00	101000.00
30	1	FLOW	1945	365	3322000	311	00000001003631000555	14530220000000000467	95800.00	40000.00	40000.00	40000.00	27000.00
31	1	FLOW	1946	365	3322000	1	00000001003632000555	14530220000000000001	138000.00	150000.00	156000.00	153000.00	141000.00
32	1	FLOW	1946	365	3322000	63	00000001003632000555	14530220000000000077	151000.00	136000.00	134000.00	139000.00	144000.00
33	1	FLOW	1946	365	3322000	125	00000001003632000555	14530220000000000175	75000.00	79800.00	85400.00	109000.00	153000.00
34	1	FLOW	1946	365	3322000	187	00000001003632000555	14530220000000000273	183000.00	165000.00	148000.00	133000.00	116000.00
35	1	FLOW	1946	365	3322000	249	00000001003632000555	14530220000000000371	317000.00	333000.00	333000.00	320000.00	294000.00
36	1	FLOW	1946	365	3322000	311	00000001003632000555	14530220000000000467	30000.00	38000.00	38000.00	38000.00	35000.00
37	1	FLOW	1947	365	3322000	1	00000001003633000555	14530220000000000001	11000.00	11000.00	11000.00	11000.00	11000.00
38	1	FLOW	1947	365	3322000	63	00000001003633000555	14530220000000000077	45000.00	45000.00	45000.00	45000.00	30000.00
39	1	FLOW	1947	365	3322000	125	00000001003633000555	14530220000000000175	303000.00	326000.00	331000.00	328000.00	319000.00
40	1	FLOW	1947	365	3322000	187	00000001003633000555	14530220000000000273	254000.00	250000.00	255000.00	267000.00	277000.00
41	1	FLOW	1947	365	3322000	249	00000001003633000555	14530220000000000371	250000.00	222000.00	197000.00	185000.00	193000.00
42	1	FLOW	1947	365	3322000	311	00000001003633000555	14530220000000000467	70000.00	70000.00	77000.00	84300.00	83600.00
43	1	FLOW	1948	365	3322000	1	00000001003634000555	14530220000000000001	17500.00	17500.00	17500.00	17500.00	17500.00
44	1	FLOW	1948	365	3322000	63	00000001003634000555	14530220000000000077	39000.00	39000.00	39000.00	39000.00	47000.00
45	1	FLOW	1948	365	3322000	125	00000001003634000555	14530220000000000175	38000.00	38000.00	38000.00	38000.00	44000.00
46	1	FLOW	1948	365	3322000	187	00000001003634000555	14530220000000000273	403000.00	365000.00	319000.00	312000.00	329000.00
47	1	FLOW	1948	365	3322000	249	00000001003634000555	14530220000000000371	53000.00	53000.00	53000.00	53000.00	53000.00
48	1	FLOW	1948	365	3322000	311	00000001003634000555	14530220000000000467	65000.00	65000.00	73400.00	83300.00	90400.00
49	1	FLOW	1949	365	3322000	1	00000001003635000555	14530220000000000001	20000.00	20000.00	20000.00	20000.00	20000.00
50	1	FLOW	1949	365	3322000	63	00000001003635000555	14530220000000000077	163000.00	186000.00	210000.00	214000.00	204000.00
51	1	FLOW	1949	365	3322000	125	00000001003635000555	14530220000000000175	553000.00	546000.00	531000.00	515000.00	488000.00
52	1	FLOW	1949	365	3322000	187	00000001003635000555	14530220000000000273	241000.00	217000.00	197000.00	184000.00	174000.00
53	1	FLOW	1949	365	3322000	249	00000001003635000555	14530220000000000371	35000.00	35000.00	35000.00	35000.00	35000.00
54	1	FLOW	1949	365	3322000	311	00000001003635000555	14530220000000000467	35000.00	35000.00	35000.00	35000.00	32000.00
55	1	FLOW	1950	365	3322000	1	00000001003636000555	14530220000000000001	23000.00	23000.00	23000.00	23000.00	23000.00
56	1	FLOW	1950	365	3322000	63	00000001003636000555	14530220000000000077	74600.00	74100.00	74400.00	81600.00	89600.00

C A T A L O G   O F   B I N A R Y   D A T A B A S E   07/19/76

DSECT	DBR	TYPE	WYR	NOD	NUSGS	SDY	LABEL WORDS		FIRST FIVE DATA WORDS				
57	1	FLOW	1950	365	3322000	125	0000000100363600055 5	1453022000000000175	473000.00	503000.00	540000.00	578000.00	621000.00
58	1	FLOW	1950	365	3322000	187	00000001003636000555	1453022000000000273	420000.00	437000.00	447000.00	433000.00	409000.00
59	1	FLOW	1950	365	3322000	249	00000001003636000555	1453022000000000371	214000.00	213000.00	217000.00	221000.00	214000.00
60	1	FLOW	1950	365	3322000	311	00000001003636000555	1453022000000000467	47000.00	47000.00	47000.00	47000.00	37000.00
61	1	FLOW	1951	365	3322000	1	00000001003637000555	1453022000000000001	60000.00	60000.00	60000.00	60000.00	60000.00
62	1	FLOW	1951	365	3322000	63	00000001003637000555	1453022000000000077	157000.00	169000.00	219000.00	281000.00	333000.00
63	1	FLOW	1951	365	3322000	125	00000001003637000555	1453022000000000175	278000.00	282000.00	302000.00	339000.00	381000.00
64	1	FLOW	1951	365	3322000	187	00000001003637000555	1453022000000000273	301000.00	323000.00	342000.00	354000.00	355000.00
65	1	FLOW	1951	365	3322000	249	00000001003637000555	1453022000000000371	82000.00	82000.00	82000.00	74600.00	77900.00
66	1	FLOW	1951	365	3322000	311	00000001003637000555	1453022000000000467	20000.00	20000.00	20000.00	20000.00	23000.00
67	1	FLOW	1952	365	3322000	1	00000001003640000555	1453022000000000001	12000.00	12000.00	12000.00	12000.00	12000.00
68	1	FLOW	1952	365	3322000	63	00000001003640000555	1453022000000000077	222000.00	208000.00	198000.00	189000.00	177000.00
69	1	FLOW	1952	365	3322000	125	00000001003640000555	1453022000000000175	527000.00	560000.00	588000.00	608000.00	608000.00
70	1	FLOW	1952	365	3322000	187	00000001003640000555	1453022000000000273	263000.00	237000.00	225000.00	221000.00	219000.00
71	1	FLOW	1952	365	3322000	249	00000001003640000555	1453022000000000371	144000.00	96200.00	45000.00	45000.00	45000.00
72	1	FLOW	1952	365	3322000	311	00000001003640000555	1453022000000000467	20000.00	20000.00	20000.00	20000.00	26000.00
73	1	FLOW	1953	365	3322000	1	00000001003641000555	1453022000000000001	13000.00	13000.00	13000.00	13000.00	13000.00
74	1	FLOW	1953	365	3322000	63	00000001003641000555	1453022000000000077	21000.00	21000.00	21000.00	21000.00	38000.00
75	1	FLOW	1953	365	3322000	125	00000001003641000555	1453022000000000175	232000.00	225000.00	216000.00	204000.00	186000.00
76	1	FLOW	1953	365	3322000	187	00000001003641000555	1453022000000000273	205000.00	187000.00	174000.00	172000.00	176000.00
77	1	FLOW	1953	365	3322000	249	00000001003641000555	1453022000000000371	120000.00	122000.00	110000.00	136000.00	130000.00
78	1	FLOW	1953	365	3322000	311	00000001003641000555	1453022000000000467	28000.00	28000.00	28000.00	28000.00	31000.00
79	1	FLOW	1954	365	3322000	1	00000001003642000555	1453022000000000001	7000.00	7000.00	7000.00	7000.00	7000.00
80	1	FLOW	1954	365	3322000	63	00000001003642000555	1453022000000000077	13000.00	13000.00	13000.00	13000.00	19000.00
81	1	FLOW	1954	365	3322000	125	00000001003642000555	1453022000000000175	112000.00	131000.00	105000.00	75000.00	39000.00
82	1	FLOW	1954	365	3322000	187	00000001003642000555	1453022000000000273	157000.00	162000.00	163000.00	156000.00	143000.00
83	1	FLOW	1954	365	3322000	249	00000001003642000555	1453022000000000371	51000.00	51000.00	51000.00	51000.00	51000.00
84	1	FLOW	1954	365	3322000	311	00000001003642000555	1453022000000000467	64000.00	64000.00	64000.00	64000.00	33000.00
85	1	FLOW	1955	365	3322000	1	00000001003643000555	1453022000000000001	21000.00	21000.00	21000.00	21000.00	21000.00
86	1	FLOW	1955	365	3322000	63	00000001003643000555	1453022000000000077	68000.00	68000.00	68000.00	68000.00	72800.00
87	1	FLOW	1955	365	3322000	125	00000001003643000555	1453022000000000175	45000.00	54000.00	65000.00	78400.00	100000.00
88	1	FLOW	1955	365	3322000	187	00000001003643000555	1453022000000000273	235000.00	207000.00	186000.00	173000.00	158000.00
89	1	FLOW	1955	365	3322000	249	00000001003643000555	1453022000000000371	57000.00	57000.00	57000.00	57000.00	57000.00
90	1	FLOW	1955	365	3322000	311	00000001003643000555	1453022000000000467	24000.00	24000.00	24000.00	24000.00	49000.00
91	1	FLOW	1956	365	3322000	1	00000001003644000555	1453022000000000001	24000.00	24000.00	24000.00	24000.00	20000.00
92	1	FLOW	1956	365	3322000	63	00000001003644000555	1453022000000000077	59000.00	59000.00	59000.00	59000.00	90200.00
93	1	FLOW	1956	365	3322000	125	00000001003644000555	1453022000000000175	242000.00	307000.00	367000.00	400000.00	415000.00
94	1	FLOW	1956	365	3322000	187	00000001003644000555	1453022000000000273	314000.00	343000.00	360000.00	371000.00	374000.00
95	1	FLOW	1956	365	3322000	249	00000001003644000555	1453022000000000371	228000.00	220000.00	206000.00	193000.00	182000.00
96	1	FLOW	1956	365	3322000	311	00000001003644000555	1453022000000000467	99300.00	86400.00	74700.00	78400.00	108000.00
97	1	FLOW	1957	365	3322000	1	00000001003645000555	1453022000000000001	39000.00	39000.00	39000.00	39000.00	39000.00
98	1	FLOW	1957	365	3322000	63	00000001003645000555	1453022000000000077	38000.00	38000.00	38000.00	38000.00	40000.00
99	1	FLOW	1957	365	3322000	125	00000001003645000555	1453022000000000175	373000.00	400000.00	424000.00	447000.00	461000.00
100	1	FLOW	1957	365	3322000	187	00000001003645000555	1453022000000000273	239000.00	325000.00	387000.00	421000.00	441000.00
101	1	FLOW	1957	365	3322000	249	00000001003645000555	1453022000000000371	60000.00	60000.00	68000.00	73100.00	70700.00
102	1	FLOW	1957	365	3322000	311	00000001003645000555	1453022000000000467	13000.00	13000.00	13000.00	13000.00	23000.00
103	1	FLOW	1958	365	3322000	1	00000001003646000555	1453022000000000001	25000.00	25000.00	25000.00	25000.00	25000.00
104	1	FLOW	1958	365	3322000	63	00000001003646000555	1453022000000000077	120000.00	113000.00	110000.00	117000.00	108000.00
105	1	FLOW	1958	365	3322000	125	00000001003646000555	1453022000000000175	260000.00	237000.00	219000.00	210000.00	214000.00
106	1	FLOW	1958	365	3322000	187	00000001003646000555	1453022000000000273	282000.00	282000.00	276000.00	282000.00	276000.00
107	1	FLOW	1958	365	3322000	249	00000001003646000555	1453022000000000371	52000.00	52000.00	52000.00	52000.00	52000.00
108	1	FLOW	1958	365	3322000	311	00000001003646000555	1453022000000000467	276000.00	283000.00	278000.00	260000.00	231000.00
109	1	FLOW	1959	365	3322000	1	00000001003647000555	1453022000000000001	46000.00	46000.00	46000.00	46000.00	46000.00
110	1	FLOW	1959	365	3322000	63	00000001003647000555	1453022000000000077	65000.00	73800.00	65000.00	70700.00	98200.00
111	1	FLOW	1959	365	3322000	125	00000001003647000555	1453022000000000175	454000.00	398000.00	351000.00	283000.00	243000.00
112	1	FLOW	1959	365	3322000	187	00000001003647000555	1453022000000000273	180000.00	178000.00	177000.00	186000.00	204000.00

```
00.23.29.      ,S24,CM15000,T60,TU5000,L10048.
00.23.29.XEQ(WABGET,ID=BXL)
00.23.30. WABGET        14 WORDS
00.23.30. PFILES COMPLETE.
00.23.31.DISABLE(PLIST)
00.37.24.PFILES(,WABLIB,ID=BXL)
00.37.26. WABLIB      1723 WORDS
00.37.26. PFILES COMPLETE.
00.37.26.FORTRAN(C)
00.38.01.CX        .149 SEC., NL  33000 WORDS
00.38.02. CTIME   0.220 SECONDS.
00.38.02.LOADX(LGO,WABLIB,USEROBJ,RUNLIB2)
00.38.04.CX        .554 SEC., NL  40000 WORDS
00.38.06.CX        .696 SEC., NL  27300 WORDS
00.38.06.DATABASE ACCESS STARTED.
00.38.08. STOP
00.38.08.MAX TRACKS:  4
00.38.08.CP SECS:       1.172 = $ 0.05
00.38.08.IO UNITS:        365 = $ 0.05
00.38.08.LINES:           171 = $ 0.03
00.38.08.TOTAL COST ESTIMATE = $ 0.13
```

<u>APPENDIX C</u> - *CONTROL CARD, PROGRAM LISTING, AND SAMPLE OUTPUT OF* TESTWAB. TESTWAB *DEMONSTRATES HOW TO CALL THE* RNDSYS *UTILITY ROUTINES AND THE DATA RETRIEVAL ROUTINES.*

```
VINTAGE RUN          00.38.01. 07/20/76.

             PROGRAM TESTWAB(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
   3         REAL B(366)
   3         INTEGER GETID
       C
       C   THIS PROGRAM WILL TEST THE VARIOUS *RNDSYS* DATABASE
       C   ROUTINES.
       C
       C   RETRIEVE AND PRINT DATA FOR FLOW DATA BLOCK 4, AND
       C   PRECIPITATION FROM DATA BLOCK 124
       C   ALSO TEST *GETID* AND *GETDA*
       C
       C
       C   CALL RANDOM -- IMPORTANT
   3         CALL RANDOM
       C
   4         ID=GETID(4,1)
       C
       C   CALL RDDATA FOR DATA BLOCK 4, YEAR 1968.
       C   TYPE IS FLOW, THUS CALL WITH POSITIVE INTEGER (1)
       C
   7         CALL RDDATA(4,1968,B,ND,1)
       C
       C   CALL GETDA TO FIND THE DRAINAGE AREA:
       C
  12         DA=GETDA(4)
       C
       C   WRITE OUT: USGS NO, WS AREA, NUMBER OF DAYS IN 1968, AND THE
       C              DATA ARRAY (B)
       C
  15         WRITE(6,10)ID,DA,ND,B
  30    10   FORMAT(≠1  USGS NO = ≠,I10,≠    DRAINAGE (WATERSHED) AREA = ≠
            *,F8.0/≠      NUMBER OF DAYS IN WATER YEAR 1968 =≠,I5//(13(1X,F9.2))
       C
       C   PRECIPITATION FROM DATA BLOCK 124:
       C   TYPE IS (0) FOR PRECIP
       C
  30         ID=GETID(124,0)
       C
  33         CALL RDDATA(124,1968,B,ND,0)
       C
       C   WRITE OUT RESULTS
       C
  36         WRITE(6,15)ID,ND,B
  50    15   FORMAT(//≠ USGS NO = ≠,I10,≠ NUMBER OF DAYS IN WATER YEAR 1968 =≠,
            *I5//(13(1X,F9.2))
       C
       C
       C
       C   NOW, JUST FOR THE HECK OF IT, LET≠S SET THE *N* AND *K* VALUES
       C   TO 0.400, AND 0.500, THEN READ THEM BACK OUT.
       C
  50         CALL SETN(4,400)
  52         CALL SETK(4,500)
       C
  54         VALUEN=GETN(4)

TESTWAB
```

APPENDIX C – TESTWAB *(cont'd)*

```
VINTAGE RUM          00.38.01. 07/20/76.

   57            VALUEK=GETK(4)
         C
   61            WRITE(6,20)VALUEN,VALUEK
   70       20   FORMAT(//≠  THE *N* VALUE = ≠,F6.4,≠   AND THE *K* VALUE = ≠,F6.4
                *///)
         C
         C
         C
         C    NOW LET≠S SEE WHAT IS IN DATA BLOCK 16 FOR WATER YEAR 1969,
         C    BUT ONLY FOR A COUPLE OF DAYS
         C
   70            CALL SETYR(1969)
         C
         C    IF SETYR WAS NOT CALLED, THE YEAR WOULD BE 1968, BECAUSE THAT
         C    IS WHAT *RDDATA* SET.
         C
         C    WE ARE INTERESTED IN DAY 1,3, AND 5.  SO...
         C
   72            DO 30 I=1,5,2
         C
         C    WE ARE INTERESTED IN FLOW DATA
         C
   74            FLOW=STREAM(16,I)
         C
   77       30   WRITE(6,40)I,FLOW
  110       40   FORMAT(≠  FOR DAY ≠,I3,≠ THE AMOUNT OF DISCHARGE WAS≠,F10.2,
                *≠ CFS≠)
         C
         C
         C
         C    ˅THAT≠S ALL FOLKS...˅
         C
         C
         C
  110            STOP
  112            END
```

APPENDIX C – TESTWAB *(cont'd)*

USGS NO = 3323000 DRAINAGE (WATERSHED) AREA = 532
NUMBER OF DAYS IN WATER YEAR 1968 = 366

8.00	8.20	11.00	9.40	11.00	8.80	8.60	9.00	8.40	8.60	11.00	12.00	12.00
12.00	14.00	13.00	22.00	20.00	21.00	19.00	24.00	21.00	18.00	18.00	16.00	16.00
18.00	18.00	18.00	19.00	19.00	26.00	38.00	74.00	158.00	142.00	127.00	99.00	67.00
49.00	38.00	35.00	34.00	28.00	24.00	25.00	24.00	36.00	48.00	68.00	95.00	82.00
65.00	50.00	43.00	39.00	39.00	37.00	31.00	29.00	29.00	27.00	146.00	1300.00	1700.00
2250.00	2700.00	2600.00	2170.00	1680.00	1480.00	2000.00	2280.00	2700.00	2640.00	2010.00	1290.00	702.00
582.00	691.00	652.00	1270.00	2340.00	2300.00	2520.00	2060.00	1250.00	597.00	412.00	362.00	350.00
320.00	310.00	280.00	270.00	260.00	250.00	240.00	230.00	220.00	220.00	220.00	210.00	210.00
210.00	200.00	200.00	200.00	200.00	200.00	200.00	200.00	459.00	958.00	1170.00	1070.00	814.00
646.00	465.00	1050.00	1870.00	2560.00	2820.00	3580.00	4860.00	4550.00	3810.00	3140.00	2050.00	1210.00
914.00	531.00	412.00	294.00	280.00	270.00	260.00	250.00	240.00	240.00	230.00	228.00	222.00
220.00	220.00	220.00	220.00	220.00	230.00	230.00	240.00	240.00	240.00	240.00	242.00	239.00
231.00	226.00	195.00	93.00	54.00	45.00	44.00	45.00	48.00	51.00	50.00	263.00	1000.00
1210.00	1090.00	754.00	519.00	531.00	564.00	555.00	1060.00	1690.00	1800.00	1840.00	1310.00	558.00
330.00	270.00	228.00	213.00	1200.00	1650.00	1830.00	1450.00	635.00	336.00	233.00	178.00	150.00
129.00	129.00	129.00	156.00	156.00	131.00	114.00	123.00	142.00	146.00	127.00	110.00	99.00
85.00	74.00	68.00	61.00	58.00	56.00	54.00	52.00	50.00	48.00	45.00	42.00	41.00
44.00	44.00	56.00	76.00	90.00	99.00	87.00	101.00	87.00	104.00	110.00	88.00	73.00
70.00	102.00	465.00	942.00	1200.00	1950.00	1890.00	2490.00	3360.00	2810.00	1970.00	1080.00	567.00
400.00	283.00	220.00	180.00	156.00	137.00	120.00	102.00	90.00	88.00	127.00	102.00	77.00
65.00	91.00	90.00	67.00	51.00	54.00	41.00	131.00	516.00	1300.00	1450.00	1570.00	1300.00
522.00	266.00	184.00	135.00	104.00	85.00	73.00	63.00	54.00	48.00	45.00	43.00	41.00
47.00	48.00	108.00	122.00	112.00	87.00	82.00	93.00	64.00	50.00	46.00	59.00	279.00
148.00	87.00	64.00	48.00	27.00	29.00	74.00	40.00	33.00	308.00	235.00	152.00	102.00
367.00	1230.00	1360.00	986.00	365.00	204.00	133.00	95.00	73.00	238.00	423.00	372.00	204.00
127.00	83.00	63.00	51.00	42.00	37.00	58.00	50.00	37.00	30.00	27.00	26.00	27.00
26.00	25.00	26.00	25.00	22.00	20.00	19.00	21.00	28.00	32.00	36.00	35.00	27.00
22.00	21.00	22.00	22.00	21.00	25.00	28.00	23.00	22.00	20.00	19.00	18.00	18.00
19.00	17.00											

USGS NO = -127482 NUMBER OF DAYS IN WATER YEAR 1968 = 366

.00	.00	.00	.00	.24	.16	.00	.00	.28	.08	.00	.00	.00
.21	.00	.60	.94	.45	.33	.00	.00	.00	.00	.00	.62	.00
.53	.05	.00	.00	.32	.07	.72	.10	.37	.00	.09	.00	.00
.00	.00	.23	.21	.00	.00	.01	.00	.51	.00	.00	.00	.08
.00	.00	.00	.05	.00	.00	.00	.00	.03	.02	.52	1.25	.00
.06	.40	.14	.00	.03	.41	.51	.70	.00	.00	.00	.00	.00
.22	.00	.00	2.21	1.21	.00	.00	.25	.03	.00	.00	.00	.00
.00	.02	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
.06	.08	.21	.00	.00	.00	.12	.00	.00	.02	.00	.00	.00
.00	.00	.36	.55	.44	.00	.43	1.55	.03	.00	.00	.00	.00
.00	.00	.00	.00	.00	.02	.00	.00	.00	.00	.00	.00	.05
.00	.00	.00	.00	.00	.00	.03	.11	.00	.02	.00	.00	.00
.00	.00	.00	.00	.19	.00	.00	.00	.08	.00	.00	.75	.00
.00	.00	.01	.13	.03	.26	.00	.00	.30	.00	.00	.00	.00
.00	.12	.00	.00	2.03	.02	.00	.00	.00	.00	.00	.00	.00
.00	.00	.50	.00	.04	.16	.01	.63	.00	.00	.01	.00	.01
.28	.23	.00	.00	.00	.00	.00	.04	.00	.00	.00	.00	.00
.24	.00	.22	.08	.00	.09	.00	1.75	.00	.00	.25	.04	.05
.02	.45	.10	.00	.04	.56	.02	.04	.13	.00	.23	.37	.00
.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.24
.00	.00	.26	.01	.00	.22	.00	1.62	1.35	.90	.03	.08	.00
.02	.72	.00	.00	.00	.02	.00	.00	.00	.00	.01	.00	.00
.00	.00	.00	.04	.00	.00	.16	.00	.00	.00	.00	2.93	.19
.00	.00	.33	.00	.00	.00	.13	.00	.00	1.30	.04	.01	.00

APPENDIX C - TESTWAB (cont'd)

.00	.03	.05	.00	.00	.00	.00	.00	1.03	1.30	.00	.00	.00
.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.54	.54
.00	.00	.50	.00	.00	.00	.10	.05	.35	.01	.00	.00	.00
.00	.18	.08	1.26	.18	.00	.00	.18	.00	.10	.00	.00	.00
.00	.00											

THE *N* VALUE = .4000 AND THE *K* VALUE = .5000

FOR DAY 1 THE AMOUNT OF DISCHARGE WAS 1270.00 CFS
FOR DAY 3 THE AMOUNT OF DISCHARGE WAS 973.00 CFS
FOR DAY 5 THE AMOUNT OF DISCHARGE WAS 1410.00 CFS

```
      PROGRAM MAIN(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE1)
      DIMENSION H(30),X(30),IA(6),E(3700)
      COMMON D(3700,6),MD,MR(6),MS(6),MODEL
C****************************************************************
C  READ IN HISTORICAL DATA FOR ROUTING MODEL , PERIOD OFESTIMATION
C  AND THE UNGAGED SIDE INFLOW CONTRIBUTING AREA
       CALL RDTAPE(1,IA,NTOTAL)  $  READ(5,100)  INI,NEND
       RKEY = 100  $  RMIN=0.0005
       READ(5,102) A1,A2       $  FACTOR=A1
       FACTOR=FACTOR*26.888888888888889
  102  FORMAT(2F5.0)
       WRITE(6,500)  A1,A2,FACTOR,MD,NTOTAL
  500 FORMAT(///20X,≠A1,A2,FACTOR,MD,NTOTAL≠,2X//20X,3F10.2,2I9)
  100  FORMAT(8I5)
C****************************************************************
C  CHECK RAINFALL SERIES FOR POSSIBILITIES OF MISSING DATA
       IZERO=0
       DO  999 N=1,NTOTAL
       IF(D(N,3).LT.30.)  GO TO 999
       IZERO = IZERO+1
       D(N,3)=0.
  999  D(N,3) = D(N,3)*FACTOR
       WRITE(6,222) IZERO
  222  FORMAT(//10X,≠****** MISSING RAINFALL COUNTS ≠,I6)
C****************************************************************
C  BEGINNING OF ESTIMATION
C  READ CONTROL KEY ≠MMM≠FOR TYPE OF ESTIMATION
C      MMM = 0  INDICATE FIRST RUN OF THE MODEL (INITIAL)
C      MMM = 1,2,...EXCEPT 5 INDICATES FINAL RUN WITH ≠MMM≠
C                    TRIAL MODEL ORDERS ( TO BE INPUT)
       READ(5,100)  MMM
        IF(MMM.GT.0) GO TO 2
C****************************************************************
C  SET UP MODEL ORDERS MS(I) FOR ≠MMM≠ = 0
        DO 4  L=1,MD
    4   MS(L)=0
        MS(1)=1
        MMM = 5
        GO TO 6
C****************************************************************
C  READ MODEL ORDER MS(I) FOR ≠MMM≠ GREATER THAN 0
    2   READ(5,100) (MS(I),I=1,MD)
    6   CONTINUE
C****************************************************************
       DO 20  J=1,MMM
        IF(MMM.NE.5)  GO TO 10
C ***************************************************************
C  SET UP MODEL ORDER MR(I)  FOR MMM = 0
        DO 8  I=1,MD
    8   MR(I)= J-1
        MR(1)=1  $  IF(J.EQ.MMM)  MR(1)=MMM
        GO TO 12
C  READ MODEL ORDER MR(I) FOR MMM.GT.0
   10   READ(5,100)  (MR(I),I=1,MD)
   12   CONTINUE
C****************************************************************
       KWR=-5  $  MODEL=1
       CALL LMODEL(H,X,IA,INI,NEND,NTOTAL,KWR,KEY,E,R2)
       IF(J.GT.1)  RKEY=R2-R1   $  R1=R2
       WRITE(6,200)  RKEY
  200  FORMAT(10X,≠********  RKEY =  ≠,F7.3)
   20  IF(RKEY.LE.RMIN)  GO TO 22
C  IF THERE IS NO SIGNIFICANT GAIN IN R2 ,THEN STOP
   22  CONTINUE
        IF(MMM.NE.5)   STOP
C  FOR INITIAL RUN TRY 3 PARAMETER MODEL ( MUSKINGKUM )
      MD=2  $  MR(1)=1  $  MR(2)=1
      CALL LMODEL(H,X,IA,INI,NEND,NTOTAL,KWR,KEY,E,R2)
      STOP  $  END
     SUBROUTINE RDTAPE(IUNIT,IA,NTOTAL)
```

<u>APPENDIX D</u> - *LISTING OF THE ALGORITHM FOR DETERMINING THE PARAMETERS OF A MULTI-INPUT LINEAR ROUTING MODEL COMPONENT.*

```
C
C  READ DATA INPUT SERIES D(N,J) FROM TAPE ≠IUNIT≠ ,NO OF SERIES
C  ≠MD≠,PERIOD OF DATA ≠NTOTAL≠ ,AND HEADING ≠IA(I)≠
      DIMENSION  IA(1)
      COMMON D(3700,6),MD,MR(6),MS(6),MODEL
    3 CONTINUE
      READ(IUNIT)   (IA(I),I=1,6),MD,NTOTAL
      IF(EOF,IUNIT)   3,4
    4 CONTINUE
      DO 1   I=1,MD
    1 READ(IUNIT)    (D(J,I),J=1,NTOTAL)
      RETURN   $    END
      SUBROUTINE  LMODEL(H,X,IA,INI,NEND,NTOTAL,KWR,KEY,E,R2)
C  SUBROUTINE TO CALL ≠LINEAR2≠ FOR PARAMETER ESTIMATION OF
C  MULTI-INPUT LINEAR ROUTING MODEL ,THEN CALL ≠MOMENT≠ TO
C  CALCULATE 4-MOMENTS OF ≠D(N,J)≠ SERIES ,AND CALL ≠FITEST≠
C  TO PERFORM WHITENESS OF RESIDUAL(ERROR)FROM LINEAR2
      DIMENSION H(1),X(1),IA(1) ,R(50),E(1)
C                                        LEDOM,)6(SM,)6(RM,DM,)6,0033(D NOMM
      COMMON D(3700,6),MD,MR(6),MS(6),MODEL
      WRITE(6,200)   (IA(I),I=1,6),MD,NTOTAL
      IB=6HMS(I)    $    WRITE(6,202)   IB,(MS(I),I=1,MD)
      IB=6HMR(I)    $    WRITE(6,202)   IB,(MR(I),I=1,MD)
C  CALL ≠SETXY≠ TO CALCULATE ≠MX≠ NO OF TERMS IN RHS.OF EQUATION
      CALL SETXY(X,Y,INI,MX,INI,NEND)
      CALL LINEAR2(H,X,MX,INI,NEND,MODEL,KWR,E,R2)
      IB=10HDATA-D          $      N=NEND-INI+1
      DO 10    J=1,MD
   10 CALL MOMENT(D(INI,J),N,H,4,2,IB,J)
  200 FORMAT(1H1//10X,6A10,2X,2I6)
  202 FORMAT(20X,A10,6I5)
      CALL FITEST(E(INI),N,50,MX,0,DOF,R,KEY)
      RETURN             $     END
      SUBROUTINE SETXY(X,Y,N,MX,INI,NEND)
C  SUBROUTINE TO SET ≠SET X≠ OR ≠VECTOR-X≠ OF DAY ≠N≠ AND THE
C  VALUE OF ≠Y≠ OF DAY ≠N≠ FROM DATA SERIE ≠D(N,J)≠
      DIMENSION  X(1)
      COMMON D(3700,6),MD,MR(6),MS(6),MODEL
      Y=D(N,1)  $  M=0    $  N1=N+1
         DO 4  K=1,MD
         J1=MS(K)+1  $   J2=MR(K)+1
         IF(J1.GT.J2)    GO TO 4
         DO 2  J=J1,J2
         M=M+1  $  X(M)=D(N1-J,K)
    2    CONTINUE
    4    CONTINUE
C  IF ≠MODEL≠ EQUAL TO ZERO ,INCLUDE INTERCEPT OR CONSTANT TERM
      IF(MODEL.NE.0)    GO TO 6
      M=M+1        $      X(M)=1.
    6 MX = M
      RETURN           $     END
      SUBROUTINE LINEAR2(H,X,MX,INI,NEND,MODEL,KWR,E,R2)
C
C   MULTIPLE REGRESSION ANALYSIS OF A LINER MODEL
C         MODEL *      YT = SUM ( H(J)*XT(J) ) FOR DAY T
C                        J = 1,2,.....MX
C         NOTE
                VALUE OF ≠YT≠ AND ≠XT(J)≠ VECTOR ARE TO
                BE SET BY SUBROUTINE ≠SETXY≠FOR DAY ≠T≠
C  INPUT
C         MX = NO OF TERM ON THE RIGHT HAND SIDE OF EQUATION
C       INI,NEND = PERIOD TO BE USED FROM DATA SERIES IN ≠SETXY≠
C       MODEL= EQUAL TO ZERO IF INTERCEPT TO BE INCLUDED
C       KWR= WRITE KEY GREATER THAN ZERO TO PRINT CORRELATION
C            MATRIX
C  OUTPUT
C         H  = LINEAR PARAMETER VECTOR CORRESPONDING TO ≠VECTOR X≠
C         E  = ERROR ARRAY OF NEND-INI+1 ≠YT-YT ESTIMATED≠
C         R2 = REGRESION COEFICIENT
C  **************************************************************
```

APPENDIX D - LISTING MILRM PARAMETER ALGORITHM (cont'd)

```
C
C          CALCULATION OF PARAMETERS H(I)                            T  -1 T
C                                                         H(I) =  (X X)  X Y
      DIMENSION  H(1),X(1),S(4),A(50,50),B(50),SS(4,3)
      DIMENSION  E(1)
C
C     *************************************************************
      WRITE(6,200)  MX,INI,NEND,MODEL
         DO 2  I=1,MX
         B(I)=0.
         DO 2  J=1,MX
    2    A(I,J)=0.
         YTY = 0.  $   HTB=0.  $  NP=NEND-INI+1
C        CALCULARE  A=XTX  AND  B=XTY
      DO 6   N=INI,NEND
         CALL SETXY( X,Y,N,MX,INI,NEND)
         DO 4  I=1,MX
             B(I) = B(I)+X(I)*Y
             DO 4  J=1,MX
    4        A(I,J) = A(I,J)+X(I)*X(J)
    6    YTY = YTY + Y*Y
         CALL MATINV(A,MX,50,50)
C        CALCULATE LINEAR PARAMETERS H(I) = A*B
         DO 10  I=1,MX
         H(I)=0.
         DO 8  J=1,MX
    8    H(I) = H(I)+A(I,J)*B(J)
   10 HTB = HTB + H(I)*B(I)
C        CALCULATE VAR(H) AND REGRESSION COEF.
      DOF = NP - MX  $  VARE=(YTY-HTB)/DOF
      IF(KWR.LE.0)  GO TO 13
      WRITE(6,202)
      DO 12  I=1,MX
   12    WRITE(6,204) (A(I,J),J=1,MX)
   13    CONTINUE
         WRITE(6,206)
         DO 14  I=1,MX
         VARH = VARE*A(I,I)  $  FR=H(I)*H(I)/VARH
   14    WRITE(6,208)  I,H(I),VARH,FR
         R2 = HTB/YTY  $   FK=MX    $  FR=DOF*R2/FK/(1.-R2)
         WRITE(6,210)  R2,HTB,YTY,FR    $  SSE = 0.
C
C     *************************************************************
             DO 16  I=1,4
C  CALCULATE MEANS OF ≠Y≠ AND ≠YE≠ (ESTIMATE)
             DO 16  J=1,3
   16        SS(I,J)=0.
      SSE = 0.     $     N2=INI+KWR    $     F=NEND-INI+1
      DO 20  N=INI,NEND
      CALL SETXY(X,Y,N,MX,INI,NEND )      $     YE = 0.
      DO 18  J=1,MX
   18 YE = YE+H(J)*X(J)
      SS(1,1)=SS(1,1)+Y  $  SS(1,2)=SS(1,2)+YE
   20 CONTINUE
      SS(1,1)=SS(1,1)/F  $  SS(1,2)=SS(1,2)/F
      SS(1,3)=SS(1,1)-SS(1,2)
C
C     *************************************************************
C  CALCULATE 4-MOMENTS OF ≠OBSERVED Y≠,≠ESTIMATED Y≠ ,AND ≠ERROR≠
         WRITE(6,220)
         DO 26   N=INI,NEND
         CALL SETXY(X,Y,N,MX,INI,NEND)  $  YE=0.
         DO 22  J=1,MX
   22 YE = YE+H(J)*X(J)
         ER=Y-YE  $  SSE=SSE+ER*ER
         E(N) = ER
         S(1)=Y-SS(1,1)  $  S(2)=YE-SS(1,2)  $  S(3)=ER-SS(1,3)
         DO 24  J=1,3
         TEP=S(J)*S(J)        $  SS(2,J)=SS(2,J)+TEP
         SS(3,J)=SS(3,J)+TEP*S(J)
```

APPENDIX D - LISTING MILRM PARAMETER ALGORITHM (cont'd)

```
   24    SS(4,J)=SS(4,J)+TEP*TEP                                          MILRM211
         IF(N.GT.N2)   GO TO 26                                           MILRM212
         WRITE(6,222)  N,(X(J),J=1,MX),Y,YE,ER                            MILRM213
   26 CONTINUE                                                            MILRM214
C                                                                         MILRM215
C     ******************************************************              MILRM216
      R2=1.-SSE/YTY     $  WRITE(6,214)  R2,SSE,YTY                       MILRM217
      WRITE(6,212)                                                        MILRM218
      DO 30  J=1,3                                                        MILRM219
      DO 28  I=2,4                                                        MILRM220
   28 SS(I,J)=SS(I,J)/F                                                   MILRM221
      SS(3,J)=SS(3,J)/SS(2,J)/SQRT(SS(2,J))                               MILRM222
   30 SS(4,J)=SS(4,J)/(SS(2,J)*SS(2,J))                                   MILRM223
      IA=6HOBS-Y          $    WRITE(6,224)  IA,(SS(I,1),I=1,4)           MILRM224
      IA=6HEST-Y          $    WRITE(6,224)  IA,(SS(I,2),I=1,4)           MILRM225
      IA=6HERROR          $    WRITE(6,224)  IA,(SS(I,3),I=1,4)           MILRM226
C                                                                         MILRM227
C     ******************************************************              MILRM228
  200 FORMAT(/20X,30(1H*)/20X,≠MULTIPLE LINEAR REGRESSION MODEL≠/30X,     MILRM229
     1 ≠BY MATRIX VERSION≠//20X,≠NO OF INDEPENDENT VARIABLES≠,I5/20X,     MILRM230
     2 ≠INITIAL DATA POINT≠,I5/20X,≠FINAL DATA POINT≠,I5/20X,             MILRM231
     3 ≠MODEL OPTION FOR INTERCEPT≠,I5/)                                  MILRM232
  202 FORMAT(/10X,≠VARIENCE-COVARIENCE MATRIX≠/)                          MILRM233
  204 FORMAT(10X,10(E10.3,1X))                                            MILRM234
  206 FORMAT(/10X,≠COMPOMENT≠,6X,≠H(I)≠,14X,≠VAR(H)≠,12X,≠F-RATIO≠/)      MILRM235
  208 FORMAT(14X, I2,3X,3(2X,E15.7))                                      MILRM236
  210 FORMAT(/10X,≠R2,SSR,SST,F-RATIO  =≠,4(2X,E13.6))                    MILRM237
  212 FORMAT(//15X,≠VARIABLE≠,7X,≠MEAN≠,9X,≠VARIENCE≠,7X,≠SKEWNESS≠,7X,   MILRM238
     1 ≠KURTOSIS≠/)                                                       MILRM239
  214 FORMAT(/10X,≠CHECK  R2,SSE,SST =  ≠,3(2X,E13.6))                    MILRM240
  220 FORMAT(//10X,≠OUTPUT TABLE = I,X(J),Y(I),YE(I),ER(I) ≠/)            MILRM241
  222 FORMAT(2X,I5,20(1X,F5.0)/(1X,20F6.0))                               MILRM242
  224 FORMAT((14X,A10,1X,4(1X,E13.6,1X))                                  MILRM243
      RETURN     $     END                                                MILRM244
      SUBROUTINE MATINV(A,N,MA,NA)                                        MILRM245
C  SUBROUTINE TO COMPUTE INVERSION OF A SQUARE MATRIX                     MILRM246
      DIMENSION A(MA,NA)                                                  MILRM247
           DO 1  K=1,N                                                    MILRM248
           DO 2  J=1,N                                                    MILRM249
    2 IF(J.NE.K)  A(K,J) = A(K,J)/A(K,K)                                  MILRM250
         A(K,K)  =  1./A(K,K)                                             MILRM251
           DO 1  I=1,N                                                    MILRM252
         IF(I.EQ.K)  GO TO 1                                              MILRM253
           DO 3  J=1,N                                                    MILRM254
    3 IF(J.NE.K)  A(I,J) = A(I,J)-A(I,K)*A(K,J)                           MILRM255
       A(I,K)=-A(I,K)*A(K,K)                                              MILRM256
    1      CONTINUE                                                       MILRM257
      RETURN   $   END                                                    MILRM258
      SUBROUTINE MOMENT(X,N,S,ID,KR,IA,IN)                                MILRM259
C  SUBROUTINE TO COMPUTE FIRST FOUR MOMENTS OF A SERIES ≠X≠               MILRM260
C        X = SERIES OF SIZE N                                             MILRM261
C        S = ARRAY OF SIZE ≠ID≠ CONTAIN MOMENTS UPTO ≠ID≠ ORDERS          MILRM262
C       KR = WRITE KEY GREATER THAN 1 ,IN WILL BE PRINTED ALSO            MILRM263
C                      EQUAL TO ZERO,NO PRINT AT ALL                      MILRM264
C       IA,IN = ALPHANUMERIC,NUMERIC VALUES TO BE PRINTED TO              MILRM265
C                IDENTIFY ≠X≠ SERIES IN OUTPUT                            MILRM266
      DIMENSION X(1),S(1)                                                 MILRM267
      DO 1 I=1,ID                                                         MILRM268
    1 S(I)=0.                                                             MILRM269
      DO 2 I=1,N                                                          MILRM270
    2 S(1)=S(1)+X(I)                                                      MILRM271
      S(1)=S(1)/FLOAT(N)                                                  MILRM272
      DO 3 I=1,N                                                          MILRM273
      TE=X(I)-S(1)   $   TEP=TE*TE                                        MILRM274
      S(2)=S(2)+TEP                                                       MILRM275
      IF(ID.EQ.2) GO TO 3                                                 MILRM276
      S(3)=S(3)+TEP*TE                                                    MILRM277
      IF(ID.EQ.3)  GO TO 3                                                MILRM278
      S(4)=S(4)+TEP*TEP                                                   MILRM279
    3 CONTINUE                                                            MILRM280
```

APPENDIX D - LISTING MILRM PARAMETER ALGORITHM (cont'd)

```
      DO 4 I=2,ID
      S(I)=S(I)/FLOAT(N)
    4 CONTINUE
      IF(ID.EQ.2) GO TO 5
      S(3)=S(3)/S(2)  /SQRT(S(2))
      IF(ID.EQ.3)  GO TO 5
      S(4)=S(4)/(S(2)*S(2))
    5    IF(KR.EQ.0)  RETURN
      IF(KR.GT.1)  GO TO 6
      WRITE(6,200)  IA,(S(I),I=1,ID)    $  RETURN
    6 WRITE(6,202)  IA,IN,(S(I),I=1,ID)
  200 FORMAT(14X,A10,1X,4(1X,E13.6,1X))
  202 FORMAT(10X,A10,I3,2X,4(1X,E13.6,1X))
      RETURN    $    END
      SUBROUTINE FITEST(A,N,K,IP,IQ,DOF,R,KEY)
C    ** SUBROUTINE TESTING WHITE NOISE OR GOODNESS OF FIT TEST  BY
C                   1)   ANDERSON ≠ S TEST
C                   2)   PORTMANTEAU TEST
C   INPUT VARIABLE
C        A = RESIDUAL SERIES TO BE TESTED OF SIZE N
C        K = MAXIMUM LAG UESED TO COMPUTE AUTOCORRELATION COEFFICIENT
C    IP,IQ = NO OF AR- AND MA- PARAMETERS
C   OUTPUT VARIABLES      R = AUTOCORRELATION COEF OF SIZE K
C                       DOF = DEGREE OF FREEDOM COMPUTED
      DIMENSION A(1),R(1),ZA(3),ZP(3),CON(3)
      DATA CON/99.,95.,90./
      DATA ZA/2.576,1.96,1.645/
      DATA ZP/2.326,1.645,1.282/
      F = N   $  PORT=0.   $  AND1=0.    $   AND2=0.
      DO 4  I=1,K
      CALL AUTOCO(A,N,I,R(I),SXY)
      IF(AND1.LT.R(I))  AND1=R(I)    $     IF(AND2.GT.R(I))  AND2=R(I)
    4 PORT = PORT+R(I)*R(I)
      PORT=PORT*F    $    DOF=K-IP-IQ
      WRITE(6,206)   K,(R(I),I=1,K)
  206 FORMAT(/10X,≠AUTOCORRELATION OF RESIDUAL≠,I5,≠ LAGS≠/
     1 (10X,10F10.5))
      WRITE(6,207)
  207 FORMAT(//10X,≠GOODNESS OF FIT TEST≠/10X,≠PERCENT≠,3X,≠ANDERSON≠,2X
     1,≠AU≠,6X,≠AL≠,4X,≠TEST≠,3X,≠PORTMAN≠,3X,≠STATISTICS≠,3X,≠DOF≠,3X,
     2 ≠CRITICAL≠,4X,≠TEST≠)
C  **************                        ********************
      KEY =0
      DO 7  I=1,3
      AU=-1./F+ZA(I)/SQRT(F)     $    AL=-1./F-ZA(I)/SQRT(F)
      PC = 0.5*(SQRT(2.*DOF-1.)+ZP(I))**2    $  IPORT=5HPASS
      IF(PORT.GT.PC)  IPORT=5HFAIL  $    IAND=5HPASS
      IF(I.EQ.1.AND.PORT.LE.PC)    KEY=999
       IF(AND1.GT.AU.OR.AND2.LT.AL)    IAND=5HFAIL
    7 WRITE(6,208)   CON(I),AU,AL,IAND,PORT,DOF,PC,IPORT
  208 FORMAT(10X,F7.2,10X,F5.3,4X,F5.3,3X,A5,11X,F10.5,3X,F3.0,2X,F10.5,
     1 3X,A5)
      RETURN
      END
      SUBROUTINE AUTOCO(X,N,L,R)
      DIMENSION X(1)
C  COMPUTE         AUTOCORRELATION COEF. WITH LAG L
      NP=N-L   $  LP=L+1   $   PN=NP   $  SUM=0.   $   SL=0.   $   SU=0.
      DO 1 I=LP,NP
    1 SUM=SUM+X(I)
       DO 2 I=1,L
    2 SL=SL+X(I)
      NP1=NP+1
      DO 3 I=NP1,N
    3 SU=SU+X(I)
      S1=SUM+SL    $    S2=SUM+SU
      S1=S1/PN
      S2=S2/PN
      SXY=SX=SY=0.
      DO 4 I=1,NP
```

APPENDIX D - <u>LISTING MILRM PARAMETER ALGORITHM</u> (cont'd)

```
      J=I+L                                                             MILRM351
      SXY=SXY+X(I)*X(J)                                                 MILRM352
      SX=SX+X(I)*X(I)                                                   MILRM353
      SY=SY+X(J)*X(J)                                                   MILRM354
    4 CONTINUE                                                          MILRM355
      SXY=SXY/PN-S1*S2                                                  MILRM356
      SX=SX/PN-S1*S1                                                    MILRM357
      SY=SY/PN-S2*S2                                                    MILRM358
      R=SXY/SQRT(SX*SY)                                                 MILRM359
      RETURN                                                            MILRM360
      END                                                               MILRM361
```

APPENDIX D - <u>LISTING MILRM PARAMETER ALGORITHM</u> (cont'd)

```
      PROGRAM MUSKIN(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE1,PLOT)
C
C  PROGRAM TO COMPUTE PREDICTED REACH OUTFLOW BY MUSKINGKUM
C  MODEL ( SEE PWRR ≠27) FOR UPPER WABASH SYSTEM IN SIMULATION
C  MODE OR IN FORCASTING MODE
C
C  CHAIPANT RUKVICHAI,CIVIL ENGINEERING,PURDUE UNIVERSITY
C  JULY 1976
C
      DIMENSION T(400),Y(400),YE(400),ER(400),IWY(2,2),IDC(3)
      DIMENSION H(20),IA(6) ,X(20)
        COMMON D(3700,6),MD,MODEL
      NTAPE = 1
C*****************************************************************
C  READ IN MODEL NUMBER ≠MODEL≠ TO BE USED≠ IN ≠QROUT≠
C   MODEL = 1,2 3,4, ,5,6
      READ(5,105) MODEL
  105 FORMAT(I1,F5.0)
C*****************************************************************
C  READ IN DATA SERIES D(N,J) FROM UNIT ≠NTAPE≠ BY ≠RDDATA≠
        CALL RDTAPE(IA,NTOTAL,NTAPE)
        MX=3
C*****************************************************************
C  READ IN PARAMETERS C2,C1,C0 OF MUSKINGKUM MODEL AND PERIOD
C  OF COMPUTATION ≠NSTART≠-≠NEND≠  AND CORRESPONDING DATE
C  IWY(2,2)
        READ(5,102)   (H(I),I=1,MX)
        READ(5,100)   NSTART,NEND
        NP=NEND-NSTART+1
        READ(5,100) ((IWY(I,J),I=1,2),J=1,2)
C****************************************************************
  100 FORMAT(10I5)
  102 FORMAT(5F5.0     )
      DO 20  KK=1,2
      IC = 2-KK
C  BEGIN OF COMPUTATION
C  IC = 0 FOR FORCASTING MODE,IC=1 FOR SIMULATION MODE
C  WRITE OUTPUT - HEADING AND DATA
         WRITE(6,200)   (IA(I),I=1,6),IC,MD,MX
         WRITE(6,202)   (I,H(I),I=1,MX)
         WRITE(6,204)   NSTART,NEND,IWY(1,1),IWY(2,1),IWY(2,2)
         WRITE(6,205)
C  PREDICT VALUE OF OUTFLOW ≠Y≠ FROM QROUT DAY BY DAY
      DO 10  N=NSTART,NEND
      I=N-NSTART+1   $   T(I)=IWY(2,1)+I-1
      YEI   =QROUT(H,MX,X,N,Y(I),YE,NSTART,IC)   $   YE(I)=YEI
      ER(I)=Y(I)-YE(I)
   10 WRITE(6,206)   N,(X(J),J=1,MX),Y(I),YE(I),ER(I)
C  COMPUTE MOMENTS OF ≠Y≠,≠YE≠,AND ERROR ≠ER≠
C  THEN CALL ≠QPLOT≠ TO PLOT OBSERVED AND PREDICTED HYDROGRAPH
C  AND ERROR ON THE TOP
        CALL MOMENT (Y,NP,X,4,1,3HY    ,NP)
         CALL MOMENT(YE,NP,X,4,1,3HYE   ,NP)
        CALL MOMENT(ER,NP,X,4,1,3HER   ,NP)
CXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
   20 CALL QPLOT(T,Y,YE,ER,NP,IA,60,IWY,IC)
      CALL PLOT(0,0,999)
  200 FORMAT(1H1//10X,6A10,7I3)
  202 FORMAT(10X,≠PARAMETER H(I)≠,I5,2X,E15.7)
  204 FORMAT(//10X,≠NSTART,NEND,IWY≠,8I5)
  206 FORMAT(2X,I5,20(1X,F5.0))
  205 FORMAT(10X,≠N,X(J),Y(N),YE(N),ER(N)≠)
      STOP    $    END
      FUNCTION QROUT(ALPHA,MX,X,N,Y,YE,NST,IC)
C  FUNCTION TO ROUT INFLOW TO OUTFLOW BY MUSKINGKUM MODEL
C  (DETAIL IN PWRR ≠ 27) THIS ALSO INCLUDE THE ADJUSTMENT
C  OF FLOW AT INPUT AND OUTPUT POINT OF THE MODEL BY ≠PROXY≠
C  PROCEDURE.
C       FIRST , U/S GAGED INFLOW IS ADJUSTED FOR REACH INFLOW
C                 D/S GAGED OUTFLOW IS ADJUSTED FOR OUTFLOW
```

<u>APPENDIX E</u> - *LISTING OF PROGRAM* MUSKIN *USED TO COMPUTE REACH OUTFLOWS FOR THE SIX RIVER COMPONENTS OF THE UPPER WABASH SIMULATION MODEL BY USING MUSKINGUM-TYPE REACH MODELS.*

```
C        SECOND , COMPUTE THE OUTFLOW
C         THIRD , COMPUTED OUTFLOW IS ADJUSTED AGAIN TO OBTAIN
C                THE COMPUTED OUTFLOW AT D/S GAGING STATION.
C
C   COMPUTATION ON DAY ≠N≠  ≠QROUT≠ WILL BE VALUE AT D/S GAGING
C   STATION/ ALPHA=PARAMETERS/ IC = COMPUTATION MODE.
      DIMENSION ALPHA(1),X(1)   ,YE(1)
      COMMON D(3700,6),MD,MODEL
      Y = D(N,1)   $   GO TO (1,2,3,4,5,6)   MODEL
C *****************   MODEL 1 **************************
    1 F1=0.3835   $   F2=0.3835
   51 X(1) = D(N,2) + D(N,6)*F1   + D(N,4)
      X(2) = D(N-1,2)+D(N-1,6)*F1+D(N-1,4)
      X(3) = D(N-1,1)-D(N-1,5)-D(N-1,6)*F2
      IF(IC.EQ.0.OR.N.LE.NST)   GO TO 12
      X(3) = YE(N-NST)-D(N-1,6)*F2-D(N-1,5)
   12 SUM=0.
      DO 14 K=1,3
   14 SUM=SUM+ALPHA(K)*X(K)
      QROUT = SUM + D(N,6)*F2 +D(N,5)
      RETURN
C ***************** MODEL 2 **************************
    2 F1=0.1804   $    F2=0.3609
  632 X(1) = D(N,2)+D(N,5)*F1
      X(2) = D(N-1,2)+D(N-1,5)*F1
      X(3) = D(N-1,1)-D(N-1,5)*F2-D(N-1,4)
      IF(IC.EQ.0.OR.N.LE.NST)   GO TO 22
      X(3) = YE(N-NST)-D(N-1,5)*F2-D(N-1,4)
   22 SUM=0.
      DO 24 K=1,3
   24 SUM=SUM+ALPHA(K)*X(K)
      QROUT = SUM + D(N,5)*F2+D(N,4)
      RETURN
C *****************   MODEL 3 *************************
    3 F1=0.1826   $   F2=0.1991
      GO TO 632
C *****************   MODEL 4 *************************
    4 F1=0.4513   $   F2=-0.1315
      X(1)=D(N,2)+D(N,4)*F1
      X(2)=D(N-1,2)+D(N-1,4)*F1
      X(3)=D(N-1,1)-D(N-1,4)*F2
      IF(IC.EQ.0.OR.N.LE.NST)   GO TO 42
      X(3)=YE(N-NST)-D(N-1,4)*F2
   42 SUM=0.
      DO 44  K=1,3
   44 SUM=SUM+ALPHA(K)*X(K)
      QROUT = SUM + D(N,4)*F2
      RETURN
C ****************** MODEL 5 *************************
    5 F1=0.6220   $   F2=0.0948
      GO TO 51
C ****************** MODEL 6 **************************
    6 F1=0.2459    $   F2=0.5531
      GO TO 632
C *****************************************************
      END
      SUBROUTINE RDTAPE(IA,NTOTAL,NTAPE)
C  SUBROUTINE TO READ DATA IN BINARY FORMAT FROM TAPE ≠NTAPE≠
C  READ HEADING-IA(6),NO OF DATA SERIES-MD,PERIOD OF DATA-NTOTAL
C  AND DATA SERIES-D(N,J)
      DIMENSION IA(1)
       COMMON D(3700,6),MD,MR(6),MS(6),MODEL
    1     CONTINUE
          READ(NTAPE)(IA(I),I=1,6),MD,NTOTAL
          IF(EOF,NTAPE)  1,  3
    3     CONTINUE
          DO 2   I=1,MD
    2     READ(NTAPE)   (D(J,I),J=1,NTOTAL)
      RETURN    $    END
       SUBROUTINE QPLOT(X,Y,YE,ER,NP,LAB,NL,IWY,IC)
```

```
C   SUBROUTINE TO PLOT FLOW HYDROGRAPHS-OBSERVED AND PREDICTED
C   HYDROGRAPHS ON THE SAME PLOT AND ERROR ON THE TOP
C                OBSERVED HYDROGRAPH - Y(I) VS.X(I)
C                PREDICTED HYDROGRAPH - YE(I) VS. X(I)
C                ERROR OF PREDICTION - ER(I) VS. X(I)
C   LAB(J)=BOTTOM LABEL/NP=NO OF POINTS/NL=NO OF CHARACTERS
C   IN LAB(J)/IWY(2,2)=TIME PERIOD TO BE LABELED/IC=MODE KEY.
      DIMENSION X(1),Y(1),YE(1),ER(1),LAB(1),IWY(2,2),LY(2),LE(2)
      DIMENSION TX(50),TY(50)
C
      DATA NX,LX,NY,LY/9,10HTIME-DAYS    ,15,10HSTREAM FLO ,5HW-CFS   /
      DATA NE,LE/15,10HERROR,Q-QE     ,    5H-CFS   /
C
C ***************************************************************
        DX=10.  $  YL=6.  $  EL=3.
      CALL PLOT(0.,0.80,-3)  $  CALL SYMBOL(0.,-.8,0.15,LAB,0.,NL)
C ***************************************************************
      XL=(X( NP)-X(1))/DX  $  NP1=NP+1  $  NP2=NP+2  $  N=XL  $  XL1=N
      IF(XL1.LT.XL)  XL=N+1
      X(NP1)=X(1)  $  X(NP2)=DX  $  CALL SCALE(Y,YL,NP,1)
      CALL SCALE(YE,YL,NP,1)  $  Y(NP1)=0.  $  YE(NP1)=0.
      IF(YE(NP2).GT.Y(NP2)) Y(NP2)=YE(NP2)
      IF(YE(NP2).LT.Y(NP2)) YE(NP2)=Y(NP2)
      CALL AXIS(0.,0.,LX,-NX,XL,0.,X(NP1),X(NP2),0)
      CALL AXIS(0.,0.,LY,NY,YL,90.,Y(NP1),Y(NP2),-1)
      CALL PLOT(0.,YL,3)  $  CALL PLOT(XL,YL,2)  $  CALL PLOT(XL,0.,2)
      CALL LINE(X,Y,NP,1,1,2)  $  CALL LINE(X,YE,NP,1,1,5)
C ***************************************************************
        N=XL  $  N1=N+1  $  N2=N+2  $  N3=N+3  $  A=Y(NP1)+Y(NP2)*YL
        DO 2  I=1,N
        TX(I)=X(NP1)+X(NP2)*FLOAT(I-1)  $ TY(I)=A
    2   CONTINUE
        TX(N1)=X(NP1)+XL*DX  $  TY(N1)=A  $  TX(N2)=X(NP1)
        TX(N3)=DX  $  TY(N2)=Y(NP1)  $  TY(N3)=Y(NP2)
        CALL LINE(TX,TY,N1,1,-1,13)
C ***************************************************************
      IE=YL+1. $ IE1=IE+1 $ IE2=IE+2 $ A=X(NP1)+X(NP2)*XL
      DO 4  I=1,IE
      TX(I)=A
    4 TY(I)=Y(NP1)+FLOAT(I-1)*Y(NP2)
      TX(IE1)=X(NP1) $ TX(IE2)=X(NP2) $ TY(IE1)=Y(NP1) $ TY(IE2)=Y(NP2)
      CALL LINE(TX,TY,IE,1,-1,15)
C ***************************************************************
        A=XL-2.  $  B=A+0.3  $  C=B+0.1   $  D=YL-0.5
        CALL SYMBOL(A,D  ,0.1,2,0.,-1)
        CALL SYMBOL(B,D  ,0.1,2,0.,-2)
        CALL SYMBOL(C,D  ,0.1,8HOBSERVED  ,0.,8)      $ D=D-0.25
      CALL SYMBOL(A,D   ,0.1,5,0.,-1)
      CALL SYMBOL(B,D   ,0.1,5,0.,-2)
      CALL SYMBOL(C,D   ,0.1,9HPREDICTED   ,0.,9)    $  D=YL-0.5
        CALL SYMBOL(0.5,D  ,0.1,10HWATER YEAR  ,0.,10)
        CALL NUMBER(1.6,D  ,0.1,IWY(1,1),0.,10HI4,1H/          )
C******************************************************
        CALL NUMBER(2.5,D  ,0.1,IWY(1,2),0.,10HI4,1H/         )
        CALL NUMBER(3.0,D  ,0.1,IWY(2,2),0.,10HI3             )
      IF(IC.EQ.0)  GO TO 6        $    D=D-0.25
      CALL SYMBOL(0.5,D,0.1,10HSIMULATION  ,0.,10)
    6 CONTINUE
C ***************************************************************
      A=YL+0.5  $  CALL PLOT(0.,A,-3)  $  CALL SCALE(ER,EL,NP,1)
      CALL AXIS(0.,0.,LX,-NX,XL,0.,X(NP1),X(NP2),0)
      CALL AXIS(0.,0.,LE,NE,EL,90.,ER(NP1),ER(NP2),-1)
      CALL PLOT(0.,EL,3)  $  CALL PLOT(XL,EL,2)  $  CALL PLOT(XL,0.,2)
      CALL LINE(X,ER,NP,1,0,0)  $  A=ER(NP1)+EL*ER(NP2)
C ***************************************************************
        DO 8  I=1,N
        TX(I)=X(NP1)+FLOAT(I-1)*DX  $  TY(I)=A
    8   CONTINUE
        TX(N1)=X(NP1)+XL*DX  $  TY(N1)=A  $  TX(N2)=X(NP1)
        TX(N3)=X(NP2)  $  TY(N2)=ER(NP1)  $  TY(N3)=ER(NP2)
```

```
         CALL LINE(TX,TY,N1,1,-1,13)
C ******************************************************************
      IE=EL+1. $ IE1=IE+1  $  IE2=IE+2  $  A=X(NP1)+X(NP2)*XL
      DO 10  I=1,IE
      TX(I)=A
   10 TY(I)=ER(NP1)+FLOAT(I-1)*ER(NP2)
       TX(IE1)=X(NP1)  $  TX(IE2)=X(NP2)  $  TY(IE1)=ER(NP1)
      TY(IE2)=ER(NP2)  $  CALL LINE(TX,TY,IE,1,-1,15)
C ******************************************************************
      XL=XL+2.5  $  YL=-.800-0.5-YL  $  CALL PLOT(XL,YL,-3)
      RETURN   $   END
      SUBROUTINE MOMENT(X,N,S,ID,KR,IA,IN)
C   SUBROUTINE TO COMPUTE MOMENTS UPTO 4-TH ORDERS OF A DATA SERIES
      DIMENSION X(1),S(1)
      DO 1 I=1,ID
    1 S(I)=0.
      DO 2 I=1,N
    2 S(1)=S(1)+X(I)
      S(1)=S(1)/FLOAT(N)
      DO 3 I=1,N
      TE=X(I)-S(1)  $  TEP=TE*TE
      S(2)=S(2)+TEP
      IF(ID.EQ.2) GO TO 3
      S(3)=S(3)+TEP*TE
      IF(ID.EQ.3)  GO TO 3
      S(4)=S(4)+TEP*TEP
    3 CONTINUE
      DO 4 I=2,ID
      S(I)=S(I)/FLOAT(N)
    4 CONTINUE
      IF(ID.EQ.2) GO TO 5
      S(3)=S(3)/S(2)  /SQRT(S(2))
      IF(ID.EQ.3)  GO TO 5
      S(4)=S(4)/(S(2)*S(2))
    5    IF(KR.EQ.0)  RETURN
      IF(KR.GT.1)  GO TO 6
      WRITE(6,200)  IA,(S(I),I=1,ID)   $  RETURN
    6 WRITE(6,202)  IA,IN,(S(I),I=1,ID)
  200 FORMAT(14X,A10,1X,4(1X,E13.6,1X))
  202 FORMAT(10X,A10,I3,2X,4(1X,E13.6,1X))
      RETURN    $    END
```

<u>APPENDIX E</u> - *LISTING OF* MUSKIN *(CONTINUED)*

```
      PROGRAM MILRM(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE1,PLOT)
C
C  PROGRAM TO COMPUTE PREDICTED REACH OUTFLOW BY MULTI-INPUT
C  LINEAR ROUTING MODEL ( MILRM ) IN SIMULATION OR FORECASTING
C  MODE
C  CHAIPANT RUKVICHAI,CIVIL ENGINEERING,PURDUE UNIVERSITY
C  JULY 1976
C
      DIMENSION T(400),Y(400),YE(400),ER(400),IWY(2,2),IDC(3)
      DIMENSION H(20),IA(6) ,X(20)
        COMMON D(3700,6),MD,MR(6),MS(6),MODEL
      NTAPE = 1
C ***************************************************
C  READ SIDE INFLOW AREA,A, THEN CALL ≠RDDATA≠TO READ IN DATA
C  SERIES D(N,J),HEADING-IA(6),NO.OF DATA SERIES-MD,AND THE
C  PERIOD POF DATA-NTOTAL FROM THE TAPE UNIT ≠NTAPE≠
          READ(5,105) A
  105     FORMAT(F5.0)
        CALL RDTAPE(IA,NTOTAL,NTAPE)
          MD=MD-1
C  CONVERT THE RAINFALL SERIES D(N,3) FROM INCHES/DAY TO CFS.
         FACTOR = A*26.888888888888889
         DO 4 N=1,NTOTAL
         IF(D(N,3).LT.30.)  GO TO 4
         D(N,3)=0.
    4    D(N,3)=D(N,3)*FACTOR
        MODEL=2  $  ICOM=1  $  NYS=0
C*****************************************************
C  READ IN THE MODEL ORSDERS MS(I) AND MR(I)
        READ(5,100)  (MS(I),I=1,MD)
        READ(5,100)  (MR(I),I=1,MD)
C*****************************************************
C  CALCULATE THE NUMBER OF PARAMETERS ≠MX≠ FROM MS(I) AMND MR(I)
C  THEN,READ IN THE LINEAR PARAMETERS ≠H(J) FROM UNIT ≠5≠
        MX=0
        DO 2  I=1,MD
        IF(MS(I).GT.MR(I))  GO TO 2
        MX=MX+MR(I)-MS(I)+1
    2   CONTINUE
        READ(5,102)  (H(I),I=1,MX)
        DO 20  M=1,ICOM
C  READ IN THE PERIOD FOR COMPUTATION TO BE MADE ≠NSTART≠-≠NEND≠
C  AND READ THE WATER YEAR AND DATE ≠IWY(2,2) CORRESPONDING
C  TO ≠NSTART≠-≠NEND≠ ABOVE
        READ(5,100)  NSTART,NEND
        NP=NEND-NSTART+1
        READ(5,100) ((IWY(I,J),I=1,2),J=1,2)
C*****************************************************
  100 FORMAT(10I5)
  102 FORMAT(5E15.7    )
      DO 20  KK=1,2
      IC = 2-KK
C  IC = 0 FOR COMPUTATION IN ≠FORCASTING ≠ MODE
C  IC = 1 FOR COMPUTATION IN ≠ SIMULATION ≠ MODE
C  WRITE OUTPUT - HEADING AND DATA
         WRITE(6,200)  (IA(I),I=1,6),IC,MD,(MS(I),I=1,MD)
          WRITE(6,201)  (MR(I),I=1,MD)
  201    FORMAT(70X,6X,6I3)
         WRITE(6,202)  (I,H(I),I=1,MX)
         WRITE(6,204)  NYS,NSTART,NEND,IWY(1,1),IWY(2,1),IWY(2,2)
         WRITE(6,205)
C*****************************************************
C  BEGIN COMPUTATION IN ≠IC≠ MODE - COMPUTE OUTFLOW DAY BY
C  DAY BY CALLING ≠QROUT≠ YE = SUM ( H(J)*X(J) )
      DO 10  N=NSTART,NEND
      I=N-NSTART+1  $  T(I)=IWY(2,1)+I-1
      YEI  =QROUT(H,MX,X,N,Y(I),YE,NSTART,IC)  $  YE(I)=YEI
      ER(I)=Y(I)-YE(I)
   10 WRITE(6,206)  N,(X(J),J=1,MX),Y(I),YE(I),ER(I)
C  COMPUTE MOMENTS OF OBSERVED ≠Y≠ , ESTIMATED ≠YE≠ ,AND
```

<u>APPENDIX F</u> - *LISTING OF PROGRAM* MILRM *THAT CAN BE USED TO COMPUTE REACH OUTFLOWS FOR REACHES IN ANY SURFACE WATER SYSTEM USING* MILRM *TYPE REACH MODELS FOR ITS REPRESENTATION.*

```
C  ERROR ≠ER≠ . THEN CALL ≠QPLOT≠ TO PLOT Y,YE AGAINST TIME
C  -DAY ≠T≠ AND ERROR ≠ER≠ AGAINST TIME-DAY ≠T≠
         CALL MOMENT (Y,NP,X,4,1,3HY   ,NP)
          CALL MOMENT(YE,NP,X,4,1,3HYE  ,NP)
         CALL MOMENT(ER,NP,X,4,1,3HER  ,NP)
   20 CALL QPLOT(T,Y,YE,ER,NP,IA,60,IWY,IC)
       CALL PLOT(0,0,999)
C************************************************************
  200 FORMAT(1H1//10X,6A10,7I3)
  202 FORMAT(10X,≠PARAMETER H(I)≠,I5,2X,E15.7)
  204 FORMAT(//10X,≠NYS,NSTART,NEND,IWY≠,8I5)
  206 FORMAT(2X,I5,20(1X,F5.0))
  205 FORMAT(10X,≠N,X(J),Y(N),YE(N),ER(N)≠)
       STOP   $   END
       FUNCTION QROUT(ALPHA,MX,X,N,Y,YE,NST,IC)
C  FUNCTION TO ROUT THE INPUTS TO OUTPUT BY MULTI INPUT
C  LINEAR ROUTING MODEL (MILRM) FOR DAY ≠N≠
C              YE   =  SUM ( H(J)*X (J) )
C               N      J          N
C
C  CALL ≠SETXY≠ TO SET VALUE ≠Y≠ AND VECTOR X(J) FOR DAY ≠N≠
C
      DIMENSION ALPHA(1),X(1)   ,YE(1)
        SUM=0.  $  CALL SETXY(X,Y,YE,N,MX,NST,IC)
         DO 2  I=1,MX
    2    SUM  =  SUM+ALPHA(I)*X(I)
         QROUT = SUM
      RETURN   $   END
      SUBROUTINE RDTAPE(IA,NTOTAL,NTAPE)
C  SUBROUTINE TO READ DATA IN BINARY FORMAT FROM TAPE ≠NTAPE≠
C  READ HEADING-IA(6),NO OF DATA SERIES-MD,PERIOD OF DATA-NTOTAL
C  AND DATA SERIES-D(N,J)
      DIMENSION IA(1)
       COMMON D(3700,6),MD,MR(6),MS(6),MODEL
    1    CONTINUE
         READ(NTAPE)(IA(I),I=1,6),MD,NTOTAL
         IF(EOF,NTAPE)  1, 3
    3    CONTINUE
         DO 2  I=1,MD
    2    READ(NTAPE)  (D(J,I),J=1,NTOTAL)
      RETURN    $   END
       SUBROUTINE QPLOT(X,Y,YE,ER,NP,LAB,NL,IWY,IC)
C  SUBROUTINE TO PLOT FLOW HYDROGRAPHS-OBSERVED AND PREDICTED
C  HYDROGRAPHS ON THE SAME PLOT AND ERROR ON THE TOP
C              OBSERVED HYDROGRAPH - Y(I) VS.X(I)
C              PREDICTED HYDROGRAPH - YE(I) VS. X(I)
C              ERROR OF PREDICTION - ER(I) VS. X(I)
C  LAB(J)=BOTTOM LABEL/NP=NO OF POINTS/NL=NO OF CHARACTERS
C  IN LAB(J)/IWY(2,2)=TIME PERIOD TO BE LABELED/IC=MODE KEY.
      DIMENSION X(1),Y(1),YE(1),ER(1),LAB(1),IWY(2,2),LY(2),LE(2)
      DIMENSION TX(50),TY(50)
C
      DATA NX,LX,NY,LY/9,10HTIME-DAYS   ,15,10HSTREAM FLO ,5HW-CFS   /
      DATA NE,LE/15,10HERROR,Q-QE     ,   5H-CFS   /
C
C ************************************************************
        DX=10.  $  YL=6.  $  EL=3.
      CALL PLOT(0.,0.80,-3)  $  CALL SYMBOL(0.,-.8,0.15,LAB,0.,NL)
C ************************************************************
      XL=(X( NP)-X(1))/DX  $  NP1=NP+1  $  NP2=NP+2  $  N=XL  $  XL1=N
      IF(XL1.LT.XL)  XL=N+1
      X(NP1)=X(1)  $  X(NP2)=DX  $  CALL SCALE(Y,YL,NP,1)
      CALL SCALE(YE,YL,NP,1)  $  Y(NP1)=0.  $  YE(NP1)=0.
      IF(YE(NP2).GT.Y(NP2)) Y(NP2)=YE(NP2)
      IF(YE(NP2).LT.Y(NP2)) YE(NP2)=Y(NP2)
      CALL AXIS(0.,0.,LX,-NX,XL,0.,X(NP1),X(NP2),0)
      CALL AXIS(0.,0.,LY,NY,YL,90.,Y(NP1),Y(NP2),-1)
      CALL PLOT(0.,YL,3)  $  CALL PLOT(XL,YL,2)  $  CALL PLOT(XL,0.,2)
      CALL LINE(X,Y,NP,1,1,2)  $  CALL LINE(X,YE,NP,1,1,5)
C ************************************************************
```

<u>APPENDIX F</u> - *LISTING OF* MILRM *(CONTINUED)*

```
      N=XL  $  N1=N+1  $  N2=N+2  $  N3=N+3  $  A=Y(NP1)+Y(NP2)*YL
      DO 2  I=1,N
      TX(I)=X(NP1)+X(NP2)*FLOAT(I-1)  $ TY(I)=A
 2    CONTINUE
      TX(N1)=X(NP1)+XL*DX  $  TY(N1)=A  $  TX(N2)=X(NP1)
      TX(N3)=DX  $  TY(N2)=Y(NP1)  $  TY(N3)=Y(NP2)
      CALL LINE(TX,TY,N1,1,-1,13)
C ****************************************************
    IE=YL+1. $ IE1=IE+1 $ IE2=IE+2 $ A=X(NP1)+X(NP2)*XL
    DO 4  I=1,IE
    TX(I)=A
  4 TY(I)=Y(NP1)+FLOAT(I-1)*Y(NP2)
    TX(IE1)=X(NP1) $ TX(IE2)=X(NP2) $ TY(IE1)=Y(NP1) $ TY(IE2)=Y(NP2)
    CALL LINE(TX,TY,IE,1,-1,15)
C ****************************************************
      A=XL-2.  $  B=A+0.3  $  C=B+0.1   $  D=YL-0.5
      CALL SYMBOL(A,D  ,0.1,2,0.,-1)
      CALL SYMBOL(B,D  ,0.1,2,0.,-2)
      CALL SYMBOL(C,D  ,0.1,8HOBSERVED  ,0.,8)      $ D=D-0.25
    CALL SYMBOL(A,D   ,0.1,5,0.,-1)
    CALL SYMBOL(B,D   ,0.1,5,0.,-2)
    CALL SYMBOL(C,D   ,0.1,9HPREDICTED   ,0.,9)     $  D=YL-0.5
      CALL SYMBOL(0.5,D  ,0.1,10HWATER YEAR  ,0.,10)
      CALL NUMBER(1.6,D  ,0.1,IWY(1,1),0.,10HI4,1H/          )
C*************************************************
      CALL NUMBER(2.5,D  ,0.1,IWY(1,2),0.,10HI4,1H/          )
      CALL NUMBER(3.0,D  ,0.1,IWY(2,2),0.,10HI3              )
    IF(IC.EQ.0)  GO TO 6       $    D=D-0.25
    CALL SYMBOL(0.5,D,0.1,10HSIMULATION  ,0.,10)
  6 CONTINUE
C ****************************************************
    A=YL+0.5  $  CALL PLOT(0.,A,-3)  $  CALL SCALE(ER,EL,NP,1)
    CALL AXIS(0.,0.,LX,-NX,XL,0.,X(NP1),X(NP2),0)
    CALL AXIS(0.,0.,LE,NE,EL,90.,ER(NP1),ER(NP2),-1)
    CALL PLOT(0.,EL,3)  $  CALL PLOT(XL,EL,2)  $  CALL PLOT(XL,0.,2)
    CALL LINE(X,ER,NP,1,0,0)  $  A=ER(NP1)+EL*ER(NP2)
C ****************************************************
      DO 8  I=1,N
      TX(I)=X(NP1)+FLOAT(I-1)*DX  $  TY(I)=A
 8    CONTINUE
      TX(N1)=X(NP1)+XL*DX  $  TY(N1)=A  $  TX(N2)=X(NP1)
      TX(N3)=X(NP2)  $  TY(N2)=ER(NP1)  $  TY(N3)=ER(NP2)
      CALL LINE(TX,TY,N1,1,-1,13)
C ****************************************************
    IE=EL+1. $ IE1=IE+1  $  IE2=IE+2  $  A=X(NP1)+X(NP2)*XL
    DO 10  I=1,IE
    TX(I)=A
 10 TY(I)=ER(NP1)+FLOAT(I-1)*ER(NP2)
     TX(IE1)=X(NP1)  $  TX(IE2)=X(NP2)  $  TY(IE1)=ER(NP1)
    TY(IE2)=ER(NP2)  $  CALL LINE(TX,TY,IE,1,-1,15)
C ****************************************************
    XL=XL+2.5  $  YL=-.800-0.5-YL  $  CALL PLOT(XL,YL,-3)
    RETURN   $   END
    SUBROUTINE SETXY(X,Y,YE,N,MX,NST,IC)
C   SUBROUTINE TO SET THE VALUE OF OUTFLOW Y AND X (J) VECTOR
C                                               N     N
C   FOR DAY ≠N≠ FROM DATA SERIES D(N,J) AND MODEL ORDERS MS(J),
C   MR(J) FOR MULTI-INPUT LINEAR ROUTING MODEL IN THE FORCASTING
C   MODE (IC=0) OR THE SIMULATION MODE (IC=1)
    DIMENSION YE(1)
    DIMENSION  X(1)
    COMMON D(3700,6),MD,MR(6),MS(6),MODEL
    Y=D(N,1)  $  M=0   $  N1=N+1
    K1=1  $  IF(IC.EQ.0)  GO TO 3
C  SET THE VECTOR X(J) FOR SIMULATION MODE FOR THE FIRST SERIES
C  D(N,1) REPLACED BY THE PREVIOUS PREDICTED ≠YE(I≠
    K1=2  $  J1=MS(1)+1  $  J2=MR(1)+1
     IF(J2.LT.J1)  GO TO 3
    DO 8  J=J1,J2
    JJ=N1-J  $  M=M+1  $  X(M)=D(JJ,1)
```

```
      IF(JJ.GE.NST)  X(M)=YE(JJ-NST+1 )
    8 CONTINUE
    3 CONTINUE
         DO 4  K=K1,MD
         J1=MS(K)+1   $   J2=MR(K)+1
         IF(J1.GT.J2)    GO TO 4
         DO 2  J=J1,J2
         M=M+1  $  X(M)=D(N1-J,K)
    2    CONTINUE
    4    CONTINUE
      IF(MODEL.NE.0)    GO TO 6
      M=M+1       $     X(M)=1.
    6 MX = M
      RETURN          $     END
      SUBROUTINE MOMENT(X,N,S,ID,KR,IA,IN)
C     SUBROUTINE TO COMPUTE MOMENTS UPTO 4-TH ORDERS OF A DATA SERIES
      DIMENSION X(1),S(1)
      DO 1 I=1,ID
    1 S(I)=0.
      DO 2 I=1,N
    2 S(1)=S(1)+X(I)
      S(1)=S(1)/FLOAT(N)
      DO 3 I=1,N
      TE=X(I)-S(1)  $  TEP=TE*TE
      S(2)=S(2)+TEP
      IF(ID.EQ.2) GO TO 3
      S(3)=S(3)+TEP*TE
      IF(ID.EQ.3)  GO TO 3
      S(4)=S(4)+TEP*TEP
    3 CONTINUE
      DO 4 I=2,ID
      S(I)=S(I)/FLOAT(N)
    4 CONTINUE
      IF(ID.EQ.2) GO TO 5
      S(3)=S(3)/S(2)  /SQRT(S(2))
      IF(ID.EQ.3)  GO TO 5
      S(4)=S(4)/(S(2)*S(2))
    5    IF(KR.EQ.0)  RETURN
      IF(KR.GT.1)  GO TO 6
      WRITE(6,200)  IA,(S(I),I=1,ID)    $  RETURN
    6 WRITE(6,202)  IA,IN,(S(I),I=1,ID)
  200 FORMAT(14X,A10,1X,4(1X,E13.6,1X))
  202 FORMAT(10X,A10,I3,2X,4(1X,E13.6,1X))
      RETURN    $    END
```

APPENDIX F - *LISTING OF* MILRM *(CONTINUED)*